

Treball de Fi de Màster

Màster Universitari en Enginyeria Industrial

Disseny, construcció i implementació d'un sistema de monitorització en continu de la qualitat de l'aire en relació al PM

MEMÒRIA

Autor: Oleguer Ribot Negrié
Director: Jesús Andrés Álvarez Florez
Convocatòria: Gener 2020



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resum

En aquest treball es busca la construcció i desenvolupament d'un sistema d'adquisició de dades per a mesurar els nivells de partícules en suspensió PM10, en relació a la qualitat de l'aire. Un dels requeriments del projecte és el de mantenir els costos de construcció al mínim possible, fent entrar el sistema en la categoria de *low-cost*, en auge avui dia.

Per a aconseguir aquesta fita, es construirà el projecte basat en un controlador Arduino, permetent un prototipatge ràpid i una construcció barata. S'adquirirà doncs un Arduino UNO, juntament amb tots els components necessaris per a la construcció del sensor. Posteriorment es programarà el codi que farà funcionar tot el conjunt, adquirint les dades que mesurarà el sensor i emmagatzemant-les en una targeta SD. Finalment, es construirà una caixa adient per a que es pugui instal·lar a la intempèrie i el sensor sigui capaç de seguir adquirint les dades.

En les següents pàgines del treball, es podrà trobar una descripció de tots aquests components, així com les seves característiques i el cost de construcció; el codi que s'ha escrit per a fer funcionar el conjunt, amb una explicació per parts; el muntatge que s'ha dut a terme per obtenir el sistema d'adquisició de dades final i les proves que s'han realitzat al llarg del procés amb l'ànim de garantir el bon funcionament d'aquest.

Paraules clau

Qualitat de l'aire, PM10, Prototipatge, Arduino, Programació, Contaminació, *Do-it-Yourself*, *Low-cost*.

Índex

1. Glossari	9
2. Prefaci	10
2.1. Motivació	10
2.2. Requeriments previs	10
3. Prefaci	11
3.1. Objectius	11
3.2. Abast	11
4. Introducció: la qualitat de l'aire	12
4.1. Principals contaminants de l'aire	13
4.2. PM10: Què són?	15
4.3. Eines actuals per al control de la qualitat de l'aire	16
4.4. Normativa de qualitat de l'aire i l'ICQA	18
4.4.1. Normativa	18
4.4.2. Índex Català de Qualitat de l'Aire (ICQA)	19
5. Components utilitzats	21
5.1. Llista de components	21
5.1.1. Arduino UNO R3	21
5.1.2. Detector de PM10 PMS7003	22
5.1.3. Mòdul rellotge RTC DS3231	23
5.1.4. Mòdul targeta SD	24
5.1.5. Protoboard Shield	25
5.1.6. Cables Dupont	26
5.1.7. Adaptador de 1,27mm a 2,54mm	26
5.1.8. Bateria Power Bank	27
5.1.9. Caixa	28
5.2. Cost dels components	29
6. Programa/Codi	30
6.1. Llibreries a incloure i definicions inicials	30
6.2. Variables inicials	31
6.3. <i>Setup</i>	32
6.4. Funcions	33
6.4.1. Adquisició de l'hora	33
6.4.2. Gravar dades a la targeta SD	35
6.5. <i>Loop</i>	37

6.6.	Funció de comunicació amb el sensor	38
6.6.1.	Exemple de lectura	40
7.	Muntatge	42
7.1.	Muntatge pas a pas.....	42
7.2.	Esquema de connexions	48
8.	Proves	49
8.1.	Capacitat de la targeta SD	49
8.2.	Capacitat de la bateria	49
8.3.	Comprovació que els braços de PVC incorporats no modifiquen les mesures	50
8.4.	Comparació de mesura amb una estació de control de qualitat de l'aire de la Generalitat de Catalunya.....	51
	Conclusions	54
	Bibliografia	55
	Annexes	56
	Annex 1: Codi	56
	Annex 2: Comparativa gràfica de la prova de l'impacte dels braços	60

Taula d'il·lustracions

Il·lustració 1. Contaminació de Barcelona, el Desembre de 2014, per Carles Ribas (Vazquez, s.f.).....	12
Il·lustració 2. Comparació de mida del material particulat	15
Il·lustració 3. Classificació de partícules en funció del seu diàmetre aerodinàmic	16
Il·lustració 4. Zonificació de Catalunya en zones de qualitat de l'aire	17
Il·lustració 5. Imatge de l'Arduino UNO R3.....	21
Il·lustració 6. Imatge del sensor PMS7003 de PLANTOWER.....	22
Il·lustració 7. Límit de mesura del sensor PMS7003 en funció del diàmetre aerodinàmic	23
Il·lustració 8. Imatge del rellotge RTC DS3231.....	23
Il·lustració 9. Imatge del mòdul per targeta SD.....	24
Il·lustració 10. Imatge de la Protoboard Shield.....	25
Il·lustració 11. Imatge dels cables Dupont.....	26
Il·lustració 12. Imatge de l'adaptador d'1,27mm a 2,54mm	26
Il·lustració 13. Imatge de la bateria Power Bank de 25000mAh	27
Il·lustració 14. Imatge de la caixa utilitzada, ja construïda.....	28
Il·lustració 15. Codi: Llibreries incloses	30
Il·lustració 16. Codi: Variables utilitzades	31
Il·lustració 17. Codi: Setup	32
Il·lustració 18. Codi: Funció adquisició de l'hora.....	33
Il·lustració 19. Codi: Funció de gravació de dades a la targeta SD.....	35
Il·lustració 20. Codi: Funció per mostrar les dades en pantalla del PC.....	35
Il·lustració 21. Codi: Loop del programa.....	37
Il·lustració 22. Codi: Funció comunicació amb el sensor (1).....	38
Il·lustració 23. Codi: Funció comunicació amb el sensor (2).....	39
Il·lustració 24. Exemple de cadena de bytes enviada pel sensor	40
Il·lustració 25. Sensor PMS7003 amb el seu cable	42
Il·lustració 26. Arduino UNO amb la Protoboard Shield muntada	43
Il·lustració 27. Conjunt de mòduls muntats a l'Arduino	43
Il·lustració 28. Cablejat entre els diferents mòduls	44
Il·lustració 29. Placa de plàstic de forma rectangular amb la placa de l'Arduino collada amb cargols	44
Il·lustració 30. Tires de protecció al darrera de la placa	45

Il·lustració 31. Perfil en forma de L que actua com a topall, enganxat a la caixa	45
Il·lustració 32. Placa de plàstic juntament amb l'Arduino, col·locat dins la caixa	46
Il·lustració 33. Forats realitzats a la caixa, on es pot observar el sensor a l'altra banda, i els braços de PVC que s'instal·laran posteriorment	46
Il·lustració 34. Instal·lació i fixació del sensor a la caixa	47
Il·lustració 35. Construcció de la caixa i instal·lació de l'Arduino completada ..	47
Il·lustració 36. Esquema de connexions entre l'Arduino i els diferents mòduls	48
Il·lustració 37. Dues caixes emprades per la prova dels braços: la de l'esquerra amb el sensor per dins i la de la dreta amb el sensor per fora	50
Il·lustració 38. Gràfica de dades obtingudes per l'estació de mesura	52
Il·lustració 39. Gràfica de dades obtingudes pel sensor construït al voltant de les 12h	53

Taula de taules

Taula 1. Zones de qualitat de l'aire i quantitat de punts de mesura	17
Taula 2. Taula d'equivalència entre valors d'immissió del contaminant i l'índex ICQA	20
Taula 3. Qualitat de l'aire segons l'índex ICQA	20
Taula 4. Característiques Arduino UNO R3.....	21
Taula 5. Característiques sensor PMS7003	22
Taula 6. Característiques rellotge RTC DS3231	24
Taula 7. Característiques Protoboard Shield	25
Taula 8. Característiques bateria VOOE	27
Taula 9. Característiques de la Caixa	28
Taula 10. Taula del cost dels components utilitzats	29
Taula 11. Taula exemple de traducció del missatge rebut pel sensor PMS7003	41
Taula 12. Informació de l'estació amb la que es comparen les dades adquirides	52
Taula 13. Dades obtingudes pel sensor construït al voltant de les 12h	53

1. Glossari

En aquest apartat s'exposen diferents acrònims, abreviacions i paraules tècniques utilitzades durant el projecte, per tal de fer més àgil la lectura del document.

- **PM:** Acrònim de Material Particulat. Es denomina material particulat a la mescla de partícules en suspensió a l'aire. La seva composició pot ser molt variada. La mesura es cataloga en funció del diàmetre aerodinàmic teòric d'aquestes partícules.
- **PM10:** Material particulat de diàmetre 10 micròmetres
- **PM2.5:** Material particulat de diàmetre 2,5 micròmetres
- **PM1.0:** Material particulat de diàmetre 1 micròmetre
- **RTC:** Acrònim de Real Time Clock. Dispositiu inclòs en un circuit integrat que manté l'hora actual.
- **Arduino:** Companyia de desenvolupament de hardware lliure que manufactura les plaques de desenvolupament.
- **Arduino UNO:** Placa de desenvolupament utilitzada en el projecte.
- **Mòdul Shield:** Tipus de mòdul per a Arduino que permet connectar-se amb aquest posant-los un damunt de l'altre, estalviant l'ús de cables entre els dos.

2. Prefaci

2.1. Motivació

L'elecció d'un projecte és sempre una bona oportunitat per a aprendre o millorar les qualitats que es tenen. Durant els anys de grau a la universitat hem profunditzat molt en diferents camps com la termodinàmica, la mecànica o la química, però poc en electrònica i programació. Tenint en compte el grau d'importància de l'electrònica present a pràcticament tots els àmbits, vaig creure molt oportú realitzar un Treball de Fi de Màster relacionat amb aquest tema. Cada vegada més es parla de mecatrònica (conjunció de la mecànica i l'electrònica) que no de les dues matèries per separat. Per tant, aprofundir en aquest aspecte és una opció recomanable.

Tanmateix, la reducció progressiva dels temps d'entrega en projectes, per tal de satisfer demandes cada vegada més variades i exigents, porten a desenvolupaments més ràpids i modulars. En aquest entorn entra en joc Arduino, permetent realitzar aplicacions electròniques de control de forma ràpida, senzilla i molt personalitzable. Per tant, el projecte era una bona oportunitat per aprendre a construir i programar un sistema de mesura controlat amb Arduino.

Finalment, la creixent importància sobre el control de les emissions i reducció d'aquestes, principalment en grans ciutats com Barcelona, ha impulsat l'extensió del control sobre els diferents paràmetres de la qualitat de l'aire. Un d'aquests és el de les PM10, partícules en suspensió de diàmetre 10 micròmetres.

Per tot això, crec que l'elecció d'aquest projecte és una bona oportunitat per ampliar els coneixements sobre aquests temes i aprendre a programar amb Arduino.

2.2. Requeriments previs

Donat que havia d'escriure el programa de control de l'Arduino, he hagut de profunditzar més en els fonaments de programació del llenguatge C++, llenguatge que utilitza el microprocessador.

3. Prefaci

3.1. Objectius

L'objectiu principal d'aquest projecte és el de la construcció d'un sistema de mesura i monitorització del material particulat, en relació a la qualitat de l'aire. Per aconseguir això, s'haurà d'adquirir els diferents components necessaris, muntar-los i programar el codi necessari per a prendre les mesures i emmagatzemar-les en un sistema d'emmagatzematge adequat. També s'haurà de tenir present l'autonomia del sistema, així com el cost de construcció i la grandària del conjunt.

Com a objectius secundaris, es busca augmentar el coneixement sobre la programació en codi C++, aprendre a construir projectes relacionats amb el control via Arduino i finalment mostrar la importància sobre la monitorització d'aquest índex de qualitat de l'aire.

3.2. Abast

L'abast del projecte és el de la construcció del sistema de mesura amb tots els components necessaris per al correcte funcionament de la presa de mesures: controlador, sensor, bateria, emmagatzematge, etc. També queda inclosa l'escriptura del programa de control dels diferents mòduls, l'adquisició de les dades del sensor i l'emmagatzematge d'aquestes.

Queda fora de l'abast, però com a possibilitat de millora i ampliació del sistema:

- Transmissió de les dades via connexió inal·làmbrica, com podria ser via Wi-Fi o a través de la xarxa telefònica utilitzant els mòduls adients.
- Millora de l'eficiència en el consum energètic del conjunt utilitzant un controlador amb microprocessador que no consumeixi tant com l'Arduino.
- Mesura d'altres índexs apart de les PM10, com podrien ser els NOx, CO i CO2, entre altres, incorporant els sensors pertinents.
- Tractament de les dades adquirides amb el sensor, tant internament com externament.

4. Introducció: la qualitat de l'aire

En els últims anys, la preocupació per la contaminació del medi ambient ha anat en augment. Catàstrofes com l'efecte hivernacle i l'inici de la desaparició de la capa d'ozó han impulsat la implementació de mesura i control dels diferents agents contaminants. Sobre aquesta línia, la degradació progressiva de la qualitat de l'aire ha sigut un dels principals focus de control i reducció de contaminants.

La població és més conscient que mai que la contaminació de l'atmosfera no es redueix només als agents que expulsen les indústries als exteriors de les ciutats, sinó que també està molt present en aquestes, principalment degut al trànsit massiu. En aquest context, des dels governs s'ha impulsat una política d'implantació de dispositius de mesura dels diferents paràmetres de la qualitat de l'aire, amb l'objectiu de mantenir un control, observar la seva evolució i poder dur a terme accions més concretes per pal·liar aquest problema.



Il·lustració 1. Contaminació de Barcelona, el Desembre de 2014, per Carles Ribas (Vazquez, s.f.)

En aquest context, han aparegut diferents iniciatives per part de la població. Una d'aquestes és la construcció del seu propi sistema de mesures de la qualitat de l'aire amb elements més a l'abast per a qualsevol persona, ja sigui per diversió, per curiositat, o per desconfiança amb la informació proporcionada pels governs. El baix cost d'aquests dispositius obre la porta a una alternativa més econòmica a la construcció de les estacions més complexes que s'instal·len actualment.

D'aquesta forma, l'ús extens d'aquesta pràctica podria proporcionar a les entitats de control de la qualitat de l'aire un número més elevat i distribuït de valors dels diferents paràmetres que s'han de mesurar. Així doncs, es podria passar dels 16 punts de mesura actuals a la ciutat de Barcelona a un nombre molt major, depenent de l'acceptació de la iniciativa entre la població. Un nombre major i més distribuït de mesures permetrien una major resolució del mapa de contaminació i detectar de forma més senzilla l'origen de major afectació.

Actualment ja existeixen diferents empreses que comercialitzen sistemes per a monitoritzar la qualitat de l'aire, així com la temperatura, la humitat, el volum de pluja o la contaminació acústica, tot inclòs en una caixa compacta. L'empresa Kunak és un d'aquests exemples.

Davant d'aquest auge, però, apareixen altres implicacions. Els investigadors alerten de la perillositat d'inundar les bases de dades de mesures imprecises. Apareix la pregunta decisiva: són fiables les mesures que realitzen els dispositius de baix cost?. Al llarg dels temps, s'han realitzat diferents estudis intentant validar o negar aquesta pregunta. Un estudi realitzat per Núria Castell, en col·laboració amb investigadors del *Norwegian Institute for Air Research* i el *Israel Institute of Technology* (Castell, y otros, 2017), afirma que l'ús d'aquests dispositius de baix cost no són aptes per a la mesura de la qualitat de l'aire en aplicacions que requereixen gran precisió, com podria ser l'aplicació del compliment de la normativa de qualitat de cada país. En canvi però, consideren que les mesures que es realitzen són prou fiables a gran escala amb l'objectiu de mesurar, a grans trets, la qualitat de l'aire en aplicacions que no requereixen gran precisió. Aplicacions com podria ser la monitorització dels paràmetres de l'aire de forma unipersonal, amb l'objectiu de realitzar un seguiment de les condicions a les que està exposada una persona, sobretot en entorns altament contaminats. Una altra bona aplicació seria la de detectar de forma més ràpida i fàcil canvis en els patrons de contaminació de les ciutats, i poder predir en la mesura del possible els episodis més crítics de contaminació.

4.1. Principals contaminants de l'aire

La qualitat de l'aire va lligada a la densitat de certs compostos químics presents a l'atmosfera. Tot i existir multitud d'agents contaminants, només alguns d'ells estan presents en suficient proporció per a que puguin ser perjudicials per a la salut o el medi ambient. Aquest components són els que generalment es mesuren amb més freqüència, per tal de realitzar un bon seguiment d'aquests. Els principals contaminants mesurats són:

- **Òxids de carboni (COx).** Els més comuns són el monòxid de carboni (CO) i el diòxid de carboni (CO₂). Aquest contaminant prové en gran mesura per l'acció humana, com la combustió de substàncies que contenen carboni tals com el carbó per a l'obtenció d'energia elèctrica o la gasolina i dièsel per a l'automoció. (Wikipedia, s.f.)

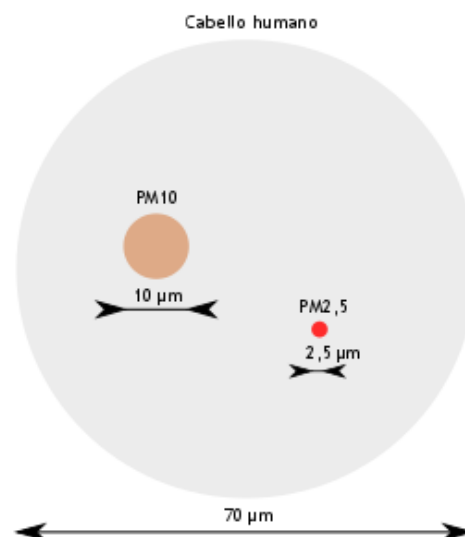
- **Òxids de nitrogen (NOx).** A diferència dels òxids de carboni, pràcticament tots els compostos derivats de la interacció entre oxigen i nitrogen es poden trobar a l'atmosfera en quantitats considerables. Aquests són N_2O , NO , N_2O_3 , N_2O_4 , NO_2 i N_2O_5 . És per aquest motiu que quan es parla d'aquests contaminants, es fa servir la paraula “nox” per a fer referència a tots ells. El procés de formació més habitual d'aquests compostos és el de la combustió a altes temperatures on l'aire és el principal comburent. (Wikipedia, s.f.)
- **Ozó (O_3).** Tot i que l'ozó és necessari en les capes altes de l'atmosfera per pal·liar l'energia de la radiació solar, en grans concentracions prop de la superfície terrestre pot ser tòxic per als éssers humans. En aquest punt, s'ha de diferenciar entre l'ozó estratosfèric (capes altes), format principalment per l'acció de la radiació ultravioleta; i l'ozó troposfèric (capes baixes) també anomenat ozó ambiental, creat a través de reaccions fotoquímiques entre els òxids de nitrogen (NOx) i els compostos orgànics volàtils derivats de la crema de combustibles. (Wikipedia, s.f.)
- **Òxids de sofre (SOx).** Aquest grup està format per SO , SO_2 i SO_3 . El diòxid de sofre (SO_2) és el més comú dels tres i el que es mesura. L'alta toxicitat i les propietats corrosives d'aquest compost el fan un dels principals enemics de la qualitat de l'aire. Comunament conegut per provocar la pluja àcida, aquest compost és alliberat en molts processos de combustió del carbó, el petroli, el dièsel o el gas natural, degut a que contenen una certa quantitat de compostos sulfurats. S'intenta per tots mitjans eliminar o reduir el màxim possible la presència d'aquests, de forma que durant la combustió no es desprenguin a l'atmosfera. (Wikipedia, s.f.)
- **Benzè.** El benzè és un compost present en molts dels productes provinents del petroli, tals com plàstics o combustibles. La perillositat d'aquest compost es presenta quan apareix en forma gas i és aspirada per l'ésser humà. S'ha demostrat que l'exposició alta a aquest contaminant pot ser molt nociu per a les persones, podent arribar a causar la mort. Els últims estudis reconeixen aquest component com a reconegut cancerigen. (Wikipedia, s.f.)
- **Partícules en suspensió o material particulat (PM10, PM2.5, PM1.0).** El material particulat està format per una sèrie de cossos sòlid o petites gotes de líquid suspeses en l'aire de l'atmosfera. La mida, composició i forma d'aquestes és variable. El rang de grandària varia entre $0,1\mu\text{m}$ i $10\mu\text{m}$. (Wikipedia, s.f.)

4.2. PM10: Què són?

Com s'ha comentat en l'apartat anterior, el material particulat fa referència a les partícules en suspensió per sota dels 10µm que es troben en l'aire. Les partícules de major diàmetre tendeixen a assentar-se més ràpidament i per tant no preocupen tant.

El principal motiu pel qual es considera crític altes densitats d'aquestes partícules és que, degut a la seva reduïda mida, poden penetrar en el sistema respiratori humà i arribar fins als pulmons, on es dipositen. Al llarg del temps, aquest fenomen comporta problemes respiratoris de tot tipus. A part del problema sanitari que comporten, també impliquen un problema ambiental.

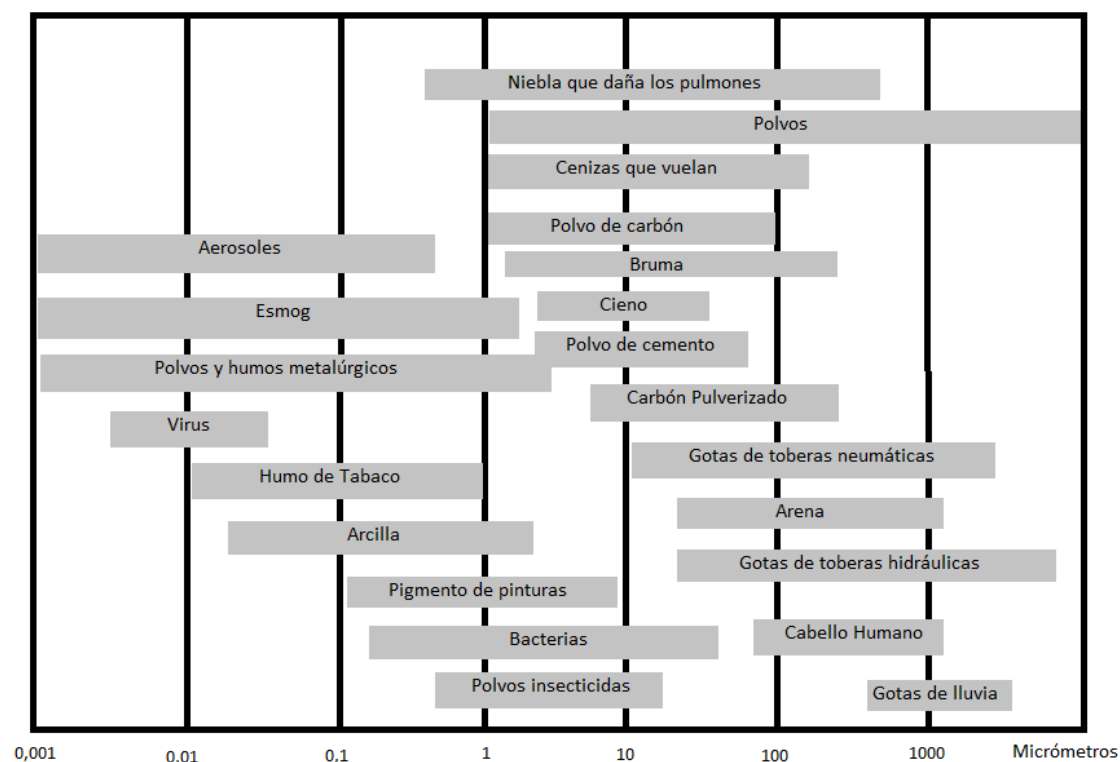
Les partícules d'aquesta mida es creen principalment per reaccions químiques i de condensació de matèria més petita, incloses molècules en estat de vapor. Una de les principals fonts de partícules atmosfèriques és la combustió incompleta de combustibles amb base carboni, com el petroli, el carbó o el dièsel, generant moltes partícules fines de sutge.



Il·lustració 2. Comparació de mida del material particulat

Les partícules en suspensió es poden classificar en funció de la seva grandària:

- PM10. Fa referència a partícules el seu diàmetre aerodinàmic és menor a 10µm. Aquest és un dels paràmetres més controlats pels diferents organismes governamentals de molts països.
- PM2.5. Anàlogament al PM10, aquest valor inclou només les partícules el seu diàmetre aerodinàmic és menor a 2,5µm. Alguns països estan passant a mesurar i tenir en compte aquest paràmetre conjuntament amb el PM10.
- PM1.0. Aquest valor inclou només les partícules el seu diàmetre aerodinàmic és menor a 1µm. Actualment no s'utilitza per a finalitats legislatives (on sí es fixen els límits de PM10 i PM2.5 permesos).



Il·lustració 3. Classificació de partícules en funció del seu diàmetre aerodinàmic

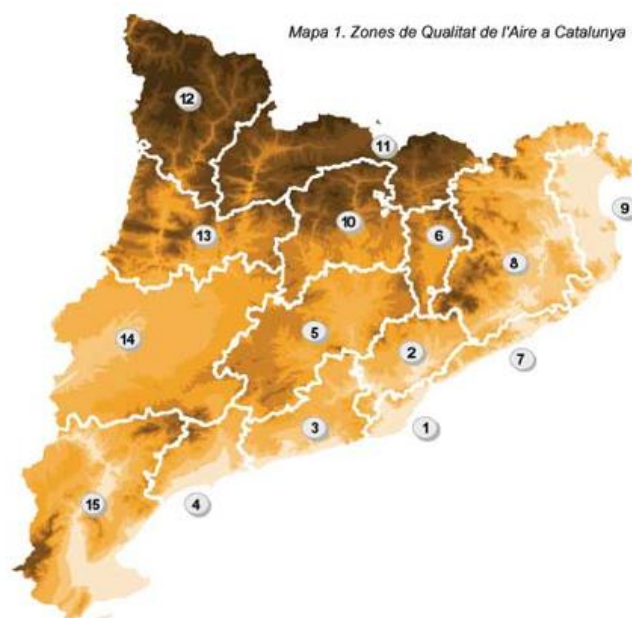
Actualment la Organització Mundial de la Salut (OMS) classifica les partícules en suspensió en gruixudes ($>2,5\mu\text{m}$) tals com la pols i terra, i en fines ($<2,5\mu\text{m}$) tals com aerosols i partícules de combustió).

4.3. Eines actuals per al control de la qualitat de l'aire

Degut als episodis cada vegada més freqüents de contaminació i a l'empitjorament de la qualitat de l'aire a les ciutats, els governs han impulsat mesures per al control i monitorització d'aquesta en les grans ciutats (Departament de Territori i Sostenibilitat, s.f.).

El Departament de Territori i Sostenibilitat del Govern de Catalunya va construir una xarxa de vigilància d'aquests paràmetres al llarg de tot el territori anomenada Xarxa de Vigilància i Previsió de la Contaminació Atmosfèrica (XVPCA). Les dades que es recullen són analitzades pel Servei de Vigilància i Control de l'Aire, i són posteriorment emmagatzemades i disposades de forma que qualsevol ciutadà pot accedir a elles de forma senzilla, en un format il·lustratiu fàcil de llegir.

L'avaluació de la qualitat de l'aire es fa d'acord amb la legislació vigent (Directiva 2008/50/CE, Reial Decret 102/2011, etc.). La legislació requereix avaluar la qualitat de l'aire per zones. Així, s'ha dividit Catalunya en 15 zones de qualitat de l'aire segons les emissions i condicions de dispersió, per tal d'optimitzar els punts de mesurament (Generalitat de Catalunya, s.f.). La distribució d'aquestes zones és la següent:



Il·lustració 4. Zonificació de Catalunya en zones de qualitat de l'aire

Número	Zona	Punts de mesura
1	Àrea de Barcelona	30
2	Vallès – Baix Llobregat	22
3	Penedès – Garraf	9
4	Camp de Tarragona	13
5	Catalunya Central	4
6	Plana de Vic	5
7	Maresme	5
8	Comarques de Girona	7
9	Empordà	2
10	Alt Llobregat	2
11	Pirineu Oriental	2
12	Pirineu Occidental	2
13	Prepirineu	2
14	Terres de Ponent	2
15	Terres de l'Ebre	14

Taula 1. Zones de qualitat de l'aire i quantitat de punts de mesura

La majoria d'aparells que hi ha instal·lats són automàtics i fixes, tot i que també es disposa de manuals i mòbils. Aquests primers, aspiren aire a cabal constant durant un temps conegut. Aquest aire es fa passar per un filtre que reté les partícules i posteriorment per un sistema de detecció diferent per a cada tipus de contaminant que es desitja mesurar. Els equips de mostreig han d'estar instal·lats en recintes amb temperatura interior més o menys constant (entre 20 i 25 graus). El període de mostreig és d'aproximadament un minut, tot i que les dades són integrades en el sistema en períodes de 10 minuts o mitja hora.

Finalment, les dades dels punts de mesurament automàtics es posen a disposició del públic a través d'una pàgina web, en la que es poden consultar les dades dels darrers sis mesos i amb una freqüència horària.

4.4. Normativa de qualitat de l'aire i l'ICQA

4.4.1. Normativa

Actualment existeix molta normativa relacionada amb la qualitat de l'aire, així com moltes entitats treballant per a que es compleixi aquesta legislació. La principal normativa relacionada és:

A nivell europeu (Ministerio para la Transición Ecológica, s.f.)

- Directiva 2008/50/CE. Introdueix regulacions per a nous contaminants, com les partícules de grandària inferior a $2,5\mu\text{m}$, conegudes també com a PM2.5, i revisa altres requisits i objectius en quant a l'avaluació de la qualitat de l'aire, tenint en compte les directrius i programes corresponents de l'Organització Mundial de la Salut.
- Directiva 2004/107/CE. Relacionat amb els contaminants com l'arsènic, el cadmi, el mercuri, el níquel i els hidrocarburs aromàtics policíclics en el medi ambient.
- Directiva 2015/1480/CE. Relativa a l'establiment de normes en quant als mètodes de referència, la validació de dades i la ubicació dels punts de mostreig per a l'avaluació de la qualitat de l'aire.

A nivell de l'Estat Espanyol (Ministerio para la Transición Ecológica, s.f.)

- Ley 34/2007. Relacionat amb l'avaluació i gestió de la qualitat de l'aire a Espanya, amb l'objectiu últim d'arribar a uns nivells òptims per evitar, prevenir o reduir riscos o efectes negatius sobre la salut humana i el medi ambient. Serveix també de marc regulador per a l'elaboració dels plans nacionals, autonòmics i locals per a la millora de la qualitat de l'aire.
- Real Decreto 102/2011. Transposa a l'ordenament jurídic espanyol el contingut de la Directiva 2008/50/CE.

A nivell de Catalunya (Departament de Territori i Sostenibilitat, s.f.)

- Llei 22/1983. Té per objectiu establir i regular els instruments i procediments per a l'actuació en el camp de la prevenció, vigilància i la correcció de la contaminació atmosfèrica (Portal Jurídic de Catalunya, s.f.).
- Decret 322/1987. Decret per al desplegament de la Llei 22/1983, esmentada anteriorment

4.4.2. [Índex Català de Qualitat de l'Aire \(ICQA\)](#)

L'ICQA és el sistema d'informació pública de l'estat de la qualitat de l'aire implantat a Catalunya des del gener de 1995. Informar dels resultats de totes les estacions de mesura i fer-ho per a cada un dels contaminants és una tasca difícil i pesada, deguda a la quantitat d'informació tècnica que es recull. Per això, aquesta informació es disposa d'una forma més pràctica per a la interpretació del gran públic (Departament de Territori i Sostenibilitat, s.f.).

L'ICQA tradueix a una mateixa escala (escala dels efectes sobre la salut de les persona) les concentracions de cada un dels contaminants mesurats. La relació entre aquest índex i els valors mesurats realment es troben a la següent taula:

	ICQA	100 a 50	49 a 0	-1 a -50	-50 a -100
Valors d'immissió	O3 1h($\mu\text{g}/\text{m}^3$)	0 – 110	111 – 180	181 – 240	> 241
	PM10 24h($\mu\text{g}/\text{m}^3$)	0 – 35	36 – 50	51 – 75	> 76
	CO 8h(mg/m^3)	0 – 5	6 – 10	11 – 15	> 16
	SO2 1h($\mu\text{g}/\text{m}^3$)	0 – 200	201 – 350	351 – 500	> 501
	NO2 1h($\mu\text{g}/\text{m}^3$)	0 – 90	91 – 200	201 – 400	> 401

Taula 2. Taula d'equivalència entre valors d'immissió del contaminant i l'índex ICQA

D'aquesta forma, es presenta de forma intuïtiva l'efecte dels nivells de contaminants mesurats, sent la qualitat de l'aire bona o dolenta si l'ICQA està entre els següents valors:

Qualitat de l'aire	Bona	Regular	Pobre
ICQA	≥ 50	0 – 49	< 0

Taula 3. Qualitat de l'aire segons l'índex ICQA

Per a obtenir el valor ICQA del dia, primerament es calcula l'ICQA a partir de la concentració de cada un dels contaminants. Posteriorment es tria l'ICQA amb valor més baix (valor més proper a zero o negatiu) i aquest serà el valor de l'ICQA d'aquell dia per aquell punt de mesurament. Per tant, val la pena remarcar que aquest índex no és una mitjana, sinó que equival a l'ICQA relatiu al contaminant que hagi produït una major afectació desfavorable sobre la qualitat de l'aire d'aquell dia.

5. Components utilitzats

5.1. Llista de components

En aquest apartat es descriuen els diferents components que s'han utilitzat en la construcció del sistema de mesura. S'ha buscat que fos el més modular possible, de forma que en cas de voler canviar un dels components o ampliar l'abast del projecte en un futur, no resulti complicat. A part, s'ha intentat que la construcció del conjunt sigui el més simplificat possible, per tal que la construcció de models similars o iguals a aquest sigui fàcil.

5.1.1. Arduino UNO R3

La placa que es requeria per al desenvolupament del projecte és la de l'empresa Arduino. De totes les opcions que la desenvolupadora de hardware ofereix, s'ha elegit el model UNO R3 en ser el més bàsic que compleix els requeriments del sistema. Depenent dels requisits de connectivitat o en cas d'ampliació del sistema de mesura, podria ser convenient canviar a una placa més avançada.



Il·lustració 5. Imatge de l'Arduino UNO R3

Les característiques principals d'aquesta placa són (Arduino, s.f.):

ARDUINO UNO R3

Microcontrolador	ATmega328P
Voltatge operatiu	5V
Voltatge recomanat	7 – 12 V
Voltatge límit	6 – 20 V
Nº pins I/O digitals	14
Nº pins I/O analògics	6
Memòria Flash	32 KB
Memòria SRAM	2 KB
Memòria EEPROM	1 KB
Velocitat de rellotge	16 MHz
Dimensions	68,6 x 53,4 mm

Taula 4. Característiques Arduino UNO R3

5.1.2. Detector de PM10 PMS7003

Per a poder mesurar les partícules, es requereix d'un sensor que ho permeti. A part, aquest sensor s'ha de poder comunicar correctament a través d'un protocol compatible amb el controlador elegit.

Després d'una cerca sobre diferents sensors compatibles amb Arduino, s'ha elegit el sensor de l'empresa PLANTOWER, model PMS7003.

Aquest model permet realitzar la mesura de partícules de fins a 0,3 micròmetres, retornant els paràmetres de PM10, PM2.5 i PM1.0. A part, és possible recuperar el nombre de partícules que ha mesurat classificant segons el diàmetre.



Il·lustració 6. Imatge del sensor PMS7003 de PLANTOWER

Les característiques principals del sensor són:

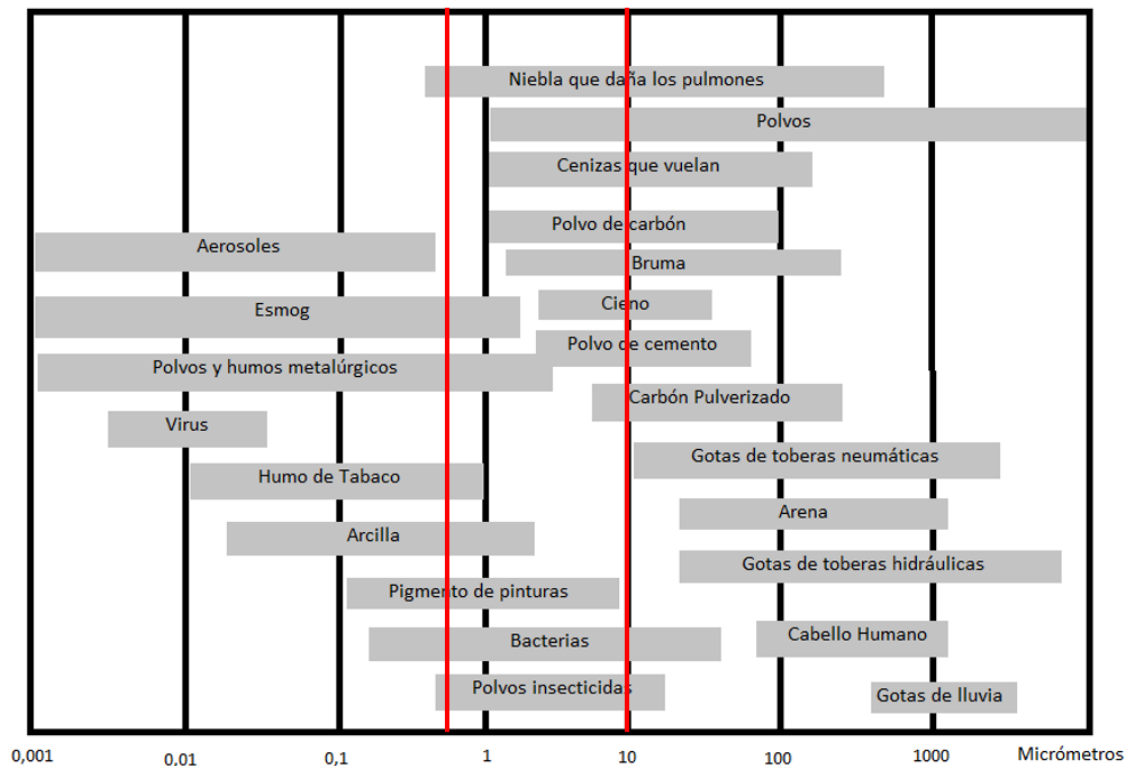
Sensor PMS7003

Rang de mesura	0,3-1,0 1,0-2,5 2,5-10 μm
Eficiència comptant partícules	0,3 μm – 50% $\geq 0,5\mu\text{m}$ – 98%
Resolució	1 $\mu\text{g}/\text{m}^3$
Volum estàndard	0,1 L
Freqüència de resposta entre mesures	<1s
Freqüència de resposta total	$\leq 10\text{s}$
Voltatge límit	4,5 – 5,5 V
Amperatge en actiu	$\leq 100\text{ mA}$
Amperatge en stand-by	$\leq 0,2\text{ mA}$
Rang de temperatura admès	-10 a +60°C
Rang d'humitat admès	0 a 99%
Temps mitjà fins la fallida (MTTF)	≥ 3 anys
Dimensions	48 x 37 x 12 mm

Taula 5. Característiques sensor PMS7003

És important tenir en compte que el sensor només és capaç de mesurar partícules de diàmetre aerodinàmic d'un rang concret i que, per tant, no mesurarà ni inclourà en els seus càlculs dels valors de PM aquestes partícules. Tal i com

s'indica en les especificacions de fàbrica, el sensor mesura partícules de $0,3\mu\text{m}$ fins a $10\mu\text{m}$. L'eficiència de comptatge de les partícules de $0,3\mu\text{m}$ però és del 50%, augmentant fins al 98% a partir de les de $0,5\mu\text{m}$. Per tant seria més pròxim a la realitat comptar amb que només és capaç de mesurar fins a $0,5\mu\text{m}$. Amb això, podem observar en la següent il·lustració que deixem de mesurar una quantitat de components importants degut a aquestes limitacions:



Il·lustració 7. Límit de mesura del sensor PMS7003 en funció del diàmetre aerodinàmic

5.1.3. Mòdul rellotge RTC DS3231

Conjuntament amb les mesures preses, es requereix saber el dia i l'hora en que s'ha pres la mesura. Tot i que l'Arduino permet adquirir l'hora actual de l'ordinador on es connecta per a transferir el programa, aquest la pot perdre en cas de desconexió de l'alimentació. Per a evitar aquest inconvenient, s'ha incorporat un rellotge extern Real Time Clock.



Il·lustració 8. Imatge del rellotge RTC DS3231

S'anomenen RTC a aquells rellotges de software que són capaços de mantenir l'hora actual, i que són diferents (i no s'han de confondre) als rellotges interns dels microprocessadors. El model DS3231 és àmpliament utilitzat per a aquest tipus d'aplicacions, sent assequible de cost i incorporant una pila per, en cas de desconnexió de l'alimentació, continuar mantenint l'hora. En cas que s'acabés també la pila, després de canviar-la es podria tornar a programar l'hora actual.

Les característiques principals del rellotge són:

Rellotge RTC DS3231

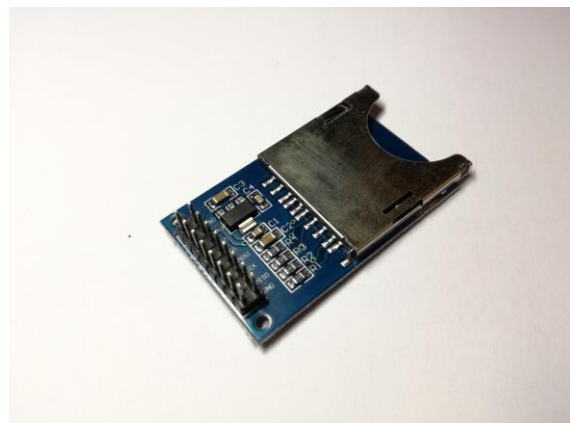
Voltatge recomanat	3,3 V
Voltatge límit	2,3 – 5,5 V
Rang de temperatura admès	-40 a 85°C
Amperatge en actiu	0,2 mA
Amperatge en stand-by	0,11 mA
Desfàs	2ppm (172ms/dia)
Tipus de pila	CR2032

Taula 6. Característiques rellotge RTC DS3231

5.1.4. Mòdul targeta SD

Degut a la quantitat de dades que s'adquireixen, és imprescindible incorporar un mètode d'emmagatzematge de les mesures. Com que la connectivitat inal·làmbrica no està contemplada en el projecte per tal d'abaratir els costos, es requereix d'un sistema amb opció de gran capacitat.

Per a aconseguir això, s'ha decidit d'incorporar un mòdul lector de targetes SD. Les targetes SD són utilitzades de forma molt comuna i permeten capacitats molt elevades. Amb aquest mateix mòdul i un adaptador, també es podria utilitzar una targeta micro-SD, avui dia casi més utilitzades que l'anterior.



Il·lustració 9. Imatge del mòdul per targeta SD

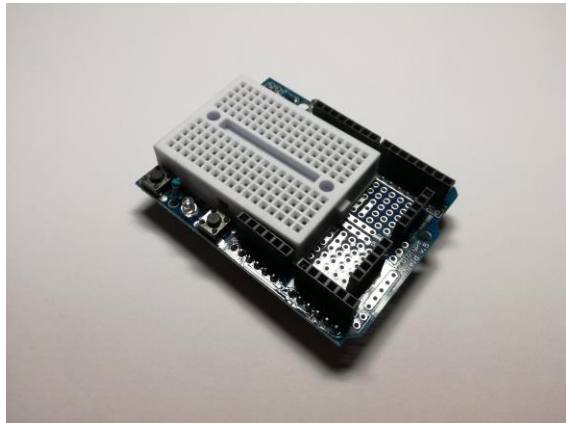
Un cop el programa adquireix les dades transmeses pel sensor, aquest obre un arxiu de text (extensió .txt) a la targeta SD, escriu les dades en una fila i finalment

guarda i tanca el fitxer. Així, per cada conjunt de dades que s'adquireix, s'accedeix una vegada a la targeta.

La tensió d'alimentació del mòdul és de 3,3V, tot i que també incorpora una entrada de 5V amb reductor de tensió.

5.1.5. Protoboard Shield

Per tal de poder connectar tot el conjunt dels mòduls, es requereix d'una placa de prototipatge, una *protoboard*. Per temes de comoditat i estètica, s'ha decidit que, en comptes d'adquirir una per separat, incloure al projecte una en format *shield*. Un *shield* és un mòdul que va connectat de forma sencera a l'Arduino, un damunt de l'altre. D'aquesta forma s'aconsegueix fer de dos mòduls separats, un conjunt més manejable.



Il·lustració 10. Imatge de la Protoboard Shield

Les característiques principals de la *protoboard* són:

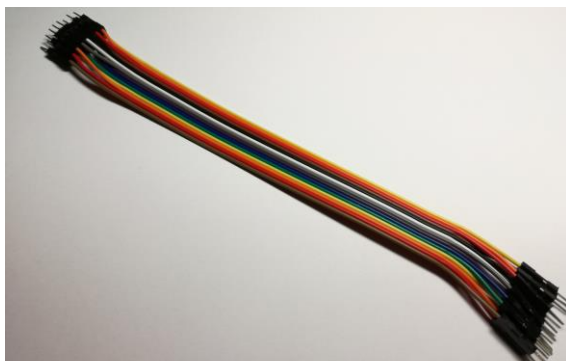
Protoboard Shield

Nº punts de connexió	34 files x 5 punts comuns
Pins 5V exposats	5
Pins GND exposats	5
Dimensions	70 x 55 x 20 mm
Altres	2 LED 1 Botó RESET 1 Botó ús general

Taula 7. Característiques Protoboard Shield

5.1.6. Cables Dupont

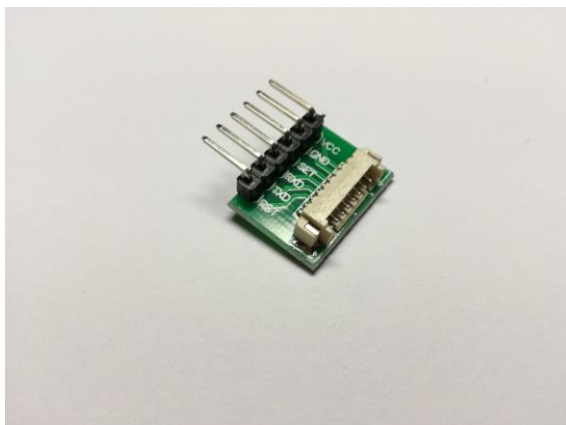
Per tal de connectar totes les entrades i sortides, es disposarà de connectors de pont per a prototips, altrament dits Dupont Cables. S'utilitzen de forma genèrica per a la construcció de prototips en plaques de prova. Tots els cables utilitzats són del tipus mascle-masclé, tot i que també existeixen femella-femella i mascle-femella.



Il·lustració 11. Imatge dels cables Dupont

5.1.7. Adaptador de 1,27mm a 2,54mm

El cable de sortida del sensor PMS7003 és un tipus de cable especial que utilitza per a transmetre les dades. Les dimensions d'aquest cable però no estan adaptats a les necessitats de la placa de prototipatge, ja que aquesta necessita cables de diàmetre 2,54mm (com els Dupont) i el cable del sensor és de 1,27mm.



Il·lustració 12. Imatge de l'adaptador d'1,27mm a 2,54mm

Per tal de realitzar les connexions de forma més senzilla, s'ha incorporat un adaptador que passa d'una mesura a l'altre, permetent connectar els cables que ja teníem.

5.1.8. Bateria Power Bank

Per a poder fer el conjunt autònom i tenir la facilitat de col·locar el sensor en l'espai que vulguem, és imprescindible que porti incorporada una bateria. Per tal d'elegir la capacitat d'aquesta i el tipus, s'han fet algunes proves. Aquestes es troben més endavant del document, a l'apartat 8. Proves.

Després de les proves, s'ha decidit que el millor tipus de bateria és una *power bank* d'Ió-Liti, amb una capacitat d'aproximadament 25.000mAh. El consum relativament elevat de l'Arduino obliga a adquirir una bateria amb una capacitat bastant gran per a tenir una autonomia decent. S'estima que amb aquesta capacitat, el sensor podria estar prenent mesures durant 12 dies de forma ininterrompuda, tot i que aquest podria variar en cas de prendre les mesures més sovint o menys.



Il·lustració 13. Imatge de la bateria Power Bank de 25000mAh

Les característiques principals de la bateria són:

Bateria VOOE

Capacitat	25.800 mAh
Port entrada	5V / 2,1A
Port sortida 1	5V / 2,1A
Port sortida 2	5V / 1A
Tipus de bateria	Polímer de liti
Dimensions	150 x 15 x 75 mm
Pes	350 g

Taula 8. Característiques bateria VOOE

5.1.9. Caixa

Finalment, es requereix d'una caixa on poder deixar tots els components per a una major comoditat i manipulació a l'hora de transportar el sensor. La caixa elegida és una caixa de connexions de plàstic. Aquest material permet aguantar correctament les condicions necessàries per a protegir el conjunt de la pluja.



Il·lustració 14. Imatge de la caixa utilitzada, ja construïda

Tot i això, s'ha hagut de practicar una obertura en aquesta per tal de treure el sensor pròpiament dit per a que pugui agafar l'aire de l'exterior. Per a resoldre els possibles problemes que podrien haver en trencar l'estanqueïtat de la caixa, s'ha optat per col·locar uns braçals de PVC enganxats a la carcassa per tal d'evitar que l'aigua de la pluja pogués entrar dins. Els detalls del muntatge es poden trobar en l'apartat 7.

Les característiques principals de la caixa són:

Caixa

Resistència aigua i pols	IP55
Resistència mecànica	IK08
Resistència al foc	HB (UL94)
Material	Plàstic
Dimensions	220 x 170 x 85 mm

Taula 9. Característiques de la Caixa

5.2. Cost dels components

El cost total dels components pot variar depenent del proveïdor o el cost d'enviament. En aquest projecte però, el cost dels diferents components és de:

Element	Quantitat	Cost Total [€]
Arduino UNO R3	1	19,90
Sensor PMS7003	1	17,00
Relloige RTC DS3231	1	7,49
Mòdul SD	1	3,99
Protoboard Shield	1	2,63
Cables Dupont	14	0,82
Adaptador	1	3,98
Bateria	1	26,95
Caixa	1	21,90
- Platina alumini plana 20cm	1	0,60
- Platina alumini forma L 20cm	1	0,96
- Perfil plàstic forma L 8cm	1	0,23
- Planxa vidre plàstic 50x50x2mm	1	3,00
- Braçal PVC Ø32mm	2	1,84
- Cargols Ø3x10mm	2	0,10
- Cargols Ø3x20mm	2	0,14
Cost Total		111,53

Taula 10. Taula del cost dels components utilitzats

6. Programa/Codi

Per a que el muntatge que s'ha realitzat funcioni, es requereix d'un programa que controli la presa de mesures, l'emmagatzemament i la gestió de l'energia. Aquesta ha sigut una de les parts més feixugues a realitzar degut a la poca experiència en el llenguatge C++.

En aquest apartat es pot trobar el codi complet utilitzat, així com l'explicació de les diferents línies de codi per a un major enteniment per al lector. Aquest apartat també pot ser útil per a possibles modificacions i millores futures. El codi sencer es pot trobar a l'Annex 1 d'aquest document.

6.1. Llibreries a incloure i definicions inicials

```
1 /*
2  Treball de Fi de Màster
3  SensorPM10
4
5  Programa per a la lectura de les mesures d'un sensor de material
6  particulat PM10, PM2.5 i PM1.0, i el posterior enregistrament de
7  les dades en una targeta SD. El programa també apunta l'hora en
8  que s'ha adquirit cada dada.
9
10 Autor: Oleguer Ribot Negrié
11 Última modificació: 10 Setembre 2019
12
13 Versió amb la sortida de dades duplicada a la pantalla d'ordinador
14 */
15
16 // Llibreries a incloure
17 #include <SPI.h>
18 #include "SdFat.h"           //Requereix instal·lar-la (Herramientas>Administrar bibliotecas...)
19 #include <Wire.h>
20 #include <RTClib.h>          //Requereix instal·lar-la (Herramientas>Administrar bibliotecas...)
21 #include <SoftwareSerial.h>
22
23
24
25 SdFat SD;
26 #define SD_CS_PIN SS
27 File arxiu;
28 RTC_DS3231 rtc;
29 SoftwareSerial pmsSerial(2, 3);
```

Il·lustració 15. Codi: Llibreries incloses

En aquesta part del codi s'inclouen en primer lloc les llibreries que necessitarà el programa:

- **SPI:** Configura i controla la comunicació del bus SPI (Serial Peripheral Interface). Utilitzada pel mòdul SD.
- **SdFat:** Incorpora diferents funcions de lectura i escriptura de dades en targetes SD. Utilitzada pel mòdul SD.
- **Wire:** Configura i controla la comunicació del bus I2C (Inter-Integrated Circuit). Utilitzada pel mòdul rellotge RTC DS3231.

- **RTCLib:** Incorpora diferents funcions per a l'adquisició de la data i l'hora, així com la configuració via software de l'hora real. Utilitzada pel mòdul rellotge RTC DS3231.
- **SoftwareSerial:** Llibreria que permet establir una comunicació en sèrie amb altres mòduls via software, utilitzant altres pins (els pins 0 i 1 queden reservats per a la comunicació amb l'ordinador, a través de "hardware serial". Per això es requereix la configuració d'altres pins per al seu ús). Utilitzada pel sensor PMS7003.

Després d'afegir les llibreries, es defineixen algunes variables i configuracions que requereixen aquestes. Les dues primeres línies les utilitza la llibreria SdFat, creant el nom de la classe que s'utilitzarà i definint en pin de control CS. La tercera crea un fitxer amb el nom 'arxiu'. La quarta la utilitza la llibreria RTCLib i defineix el nom de la classe. La última configura el nom i els pins utilitzats en la comunicació en sèrie, fent ús de la llibreria SoftwareSerial.

6.2. Variables inicials

```

33 // Variables necessàries
34 struct pms7003data {
35     uint16_t framelen;
36     uint16_t pm10_standard, pm25_standard, pm100_standard;
37     uint16_t pm10_env, pm25_env, pm100_env;
38     uint16_t particles_03um, particles_05um, particles_10um, particles_25um, particles_50um, particles_100um;
39     uint16_t unused;
40     uint16_t checksum;
41 };
42 struct pms7003data data;
43
44
45 uint8_t dormir[] = { 0x42, 0x4D, 0xE4, 0x00, 0x00, 0x01, 0x73 };
46 uint8_t despertar[] = { 0x42, 0x4D, 0xE4, 0x00, 0x01, 0x01, 0x74 };
47 boolean CHECK = false;

```

Il·lustració 16. Codi: Variables utilitzades

Aquí es defineixen les variables que es necessitaran en el programa. La primera defineix una estructura de dades que s'utilitzarà posteriorment per tal d'emmagatzemar totes les dades que provenen del sensor, i mostrar-les de forma més senzilla. Dins de l'estructura de dades, trobem les diferents variables on guardarem cada un d'aquests nombres. L'ordre ve determinat per la cadena de bytes que arriba del sensor un cop s'ha fet la mesura: primer la longitud de la cadena (frame-length), posteriorment totes les dades que es recullen (PM10, PM2.5, PM1.0) i finalment el byte de comprovació (checksum). El per què d'això s'explicarà més endavant, en l'apartat de codi que gestiona la rebuda de dades per part del sensor.

A continuació trobem tres variables més. Les dues primeres són un conjunt de bytes de configuració del mode del sensor que s'utilitzarà per tal de parar-lo i tornar-lo a posar en marxa per estalviar energia. Aquests valors es poden trobar al manual del fabricant. La tercera variable és un booleà de comprovació que s'utilitza com a validació en la rebuda de dades.

6.3. Setup

```
51 //Inicialitzacions necessàries per al programa
52 void setup() {
53   Serial.begin(9600);
54   pmsSerial.begin(9600);
55
56
57   // Comprovar que la connexió amb la SD sigui correcta (pin 4)
58   Serial.print(F("Iniciant SD...  "));
59   if (!SD.begin(SD_CS_PIN)) {
60     Serial.println(F("Error en iniciar"));
61     Serial.println(F("Comprovi que està ben connectada i les connexions ben fetes"));
62     while (1);
63   }
64   Serial.println(F("Inici de la SD correcte"));
65
66
67   //Comprovar que la connexió amb el rellotge RTC sigui correcte
68   Serial.print(F("Iniciant el RTC...  "));
69   if (!rtc.begin()) {
70     Serial.println(F("No s'ha trobat el mòdul RTC"));
71     while (1);
72   }
73   Serial.println(F("Inici del RTC correcte"));
74
75
76   //En cas que s'hagi perdut l'alimentació, fixar l'hora actual
77   if (rtc.lostPower()) {
78     rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
79   }
80 }
```

Il·lustració 17. Codi: Setup

Un cop definit tot el que serà necessari, és moment de configurar i inicialitzar els diferents mòduls. Estant els paràgrafs separats per mòduls trobem:

- 1) Inicialització de la comunicació en sèrie tant amb l'ordinador (línia 53); com amb el sensor, fent ús de la ja esmentada llibreria SerialSoftware (línia 54). Entre parèntesi trobem el “baud rate”, la velocitat a la que transferim la informació, sent de 9600 bits per segon. S'elegeix aquest número ja que és amb el que treballa el sensor.
- 2) En el segon paràgraf es comprova que el mòdul de la SD funcioni correctament. Si no es detecta, el programa llança un error en la pantalla de l'ordinador. En cas contrari, continua.

- 3) En el tercer paràgraf novament es realitza el mateix procés que en el segon, però aquesta vegada per al mòdul del rellotge.
- 4) Finalment, en cas que el rellotge hagi perdut l'alimentació (tant de l'Arduino com de la pila que incorpora), s'ajusta l'hora d'aquest de forma manual. En aquest escenari, s'agafa l'hora de l'ordinador en què estigui connectat, així que si mai s'esgotés la pila, primer s'hauria de canviar, després connectar l'Arduino a l'ordinador i finalment carregar de nou el programa i executar-lo. D'aquesta forma el rellotge tornaria a estar en hora.

6.4. Funcions

Les funcions defineixen les accions que es necessitaran realitzar durant l'execució del programa, podent-les invocar en quan es requereixi. En el codi principalment hi ha tres funcions: una per adquirir l'hora del rellotge, una altra per a escriure les dades que reben a la targeta SD i mostrar-les també a l'ordinador, i finalment una que ens permeti transformar els bytes que rebem del sensor en les dades que gravarem. Aquesta última però, s'explicarà de forma separada en un altre apartat, en ser menys intuïtiva que les dues primeres.

6.4.1. Adquisició de l'hora

```
84 // Definició de les funcions necessàries
85 void ObtenirDataHora (DateTime Temps) {
86     // Imprimir al .txt de la SD
87     arxiu.print(Temps.day());
88     arxiu.print("/");
89     arxiu.print(Temps.month());
90     arxiu.print("/");
91     arxiu.print(Temps.year());
92     arxiu.print(" ");
93     arxiu.print(Temps.hour());
94     arxiu.print(":");
95     arxiu.print(Temps.minute());
96     arxiu.print(":");
97     arxiu.print(Temps.second());
98     arxiu.print(" ");
99
100    // Imprimir a la pantalla de l'ordinador
101    Serial.print(Temps.day());
102    Serial.print(F("/"));
103    Serial.print(Temps.month());
104    Serial.print(F("/"));
105    Serial.print(Temps.year());
106    Serial.print(F(" "));
107    Serial.print(Temps.hour());
108    Serial.print(F(":"));
109    Serial.print(Temps.minute());
110    Serial.print(F(":"));
111    Serial.println(Temps.second());
112 }
```

Il·lustració 18. Codi: Funció adquisició de l'hora

En aquesta funció, s'obté el dia i l'hora en la que s'ha pres la mesura utilitzant el rellotge que s'ha incorporat al projecte. D'aquesta forma serà més senzill ubicar les dades i poder-les ordenar de forma cronològica, així com la futura comparació entre franges horàries. S'ha creat la funció per facilitar i permetre una major modularitat del codi, de forma que s'entengui millor. Aquest fragment de codi podria estar escrit directament en el cos que s'executa contínuament del programa, però s'ha preferit optar per aquesta forma en cas que es volgués modificar algun component concret del programa.

El primer paràgraf de la funció (línies 87 a 98), adquireixen el dia i l'hora actuals i la copien directament a l'arxiu que haurem creat i obert a la targeta SD. El format d'aquesta serà "dia/mes/any hora:minut:segon". Val la pena esmentar que quan coincideix un valor inferior a 10 (per exemple els segons de l'1 al 9), aquests apareixen en format d'una xifra, sense un zero davant. Així, el segon número 2 apareixerà com "2" i no "02". Aquest fet no té cap implicació per al futur tractament de dades.

El segon paràgraf de la funció (línies 101 a 111) té la mateixa funcionalitat que el primer, però aquesta vegada s'imprimeix aquesta hora en la consola de l'ordinador, en cas que l'Arduino estigui connectat a ell. D'aquesta forma es permet comprovar de forma senzilla què és el que està executant i adquirint l'Arduino. Aquest fragment de codi està especialment pensat per a fer proves i modificacions, ja que l'ús que se'n farà del sensor haurà de ser autònom, sense necessitat que estigui endollat a cap ordinador. Simplement s'executaran les línies de codi sense cap comunicació.

6.4.2. Gravar dades a la targeta SD

```
115 void ObtenirDades () {
116     if (readPMSdata(&pmsSerial)) {
117         //Obtenir la data i l'hora
118         ObtenirDataHora(rtc.now());
119         // Imprimir al .txt de la SD
120         arxiu.print(data.pm10_standard);    //Concentració PM1.0 (ug/m3) [Std]
121         arxiu.print(" ");
122         arxiu.print(data.pm25_standard);    //Concentració PM2.5 (ug/m3) [Std]
123         arxiu.print(" ");
124         arxiu.print(data.pm100_standard);   //Concentració PM10 (ug/m3) [Std]
125         arxiu.print(" ");
126         arxiu.print(data.pm10_env);         //Concentració PM1.0 (ug/m3) [Atm]
127         arxiu.print(" ");
128         arxiu.print(data.pm25_env);         //Concentració PM2.5 (ug/m3) [Atm]
129         arxiu.print(" ");
130         arxiu.print(data.pm100_env);        //Concentració PM10 (ug/m3) [Atm]
131         arxiu.print(" ");
132         arxiu.print(data.particles_03um);   //Nº Partícules >0.3um en 0.1L
133         arxiu.print(" ");
134         arxiu.print(data.particles_05um);   //Nº Partícules >0.5um en 0.1L
135         arxiu.print(" ");
136         arxiu.print(data.particles_10um);   //Nº Partícules >1.0um en 0.1L
137         arxiu.print(" ");
138         arxiu.print(data.particles_25um);   //Nº Partícules >2.5um en 0.1L
139         arxiu.print(" ");
140         arxiu.print(data.particles_50um);   //Nº Partícules >5.0um en 0.1L
141         arxiu.print(" ");
142         arxiu.println(data.particles_100um); //Nº Partícules >10um en 0.1L
```

Il·lustració 19. Codi: Funció de gravació de dades a la targeta SD

```
143
144     // Imprimir a la pantalla de l'ordinador
145     Serial.print(F("Concentració PM1.0 (ug/m3) [Std]:"));
146     Serial.println(data.pm10_standard);
147     Serial.print(F("Concentració PM2.5 (ug/m3) [Std]:"));
148     Serial.println(data.pm25_standard);
149     Serial.print(F("Concentració PM10 (ug/m3) [Std]:"));
150     Serial.println(data.pm100_standard);
151     Serial.print(F("Concentració PM1.0 (ug/m3) [Atm]:"));
152     Serial.println(data.pm10_env);
153     Serial.print(F("Concentració PM2.5 (ug/m3) [Atm]:"));
154     Serial.println(data.pm25_env);
155     Serial.print(F("Concentració PM10 (ug/m3) [Atm]:"));
156     Serial.println(data.pm100_env);
157     Serial.print(F("Nº Partícules >0.3um en 0.1L:"));
158     Serial.println(data.particles_03um);
159     Serial.print(F("Nº Partícules >0.5um en 0.1L:"));
160     Serial.println(data.particles_05um);
161     Serial.print(F("Nº Partícules >1.0um en 0.1L:"));
162     Serial.println(data.particles_10um);
163     Serial.print(F("Nº Partícules >2.5um en 0.1L:"));
164     Serial.println(data.particles_25um);
165     Serial.print(F("Nº Partícules >5.0um en 0.1L:"));
166     Serial.println(data.particles_50um);
167     Serial.print(F("Nº Partícules >10um en 0.1L:"));
168     Serial.println(data.particles_100um);
169     }
170 }
```

Il·lustració 20. Codi: Funció per mostrar les dades en pantalla del PC

L'objectiu d'aquesta funció és la de cridar la funció d'adquisició de l'hora, i posteriorment gravar les mesures que just s'han realitzat a la targeta SD. Novament com en la funció anterior, es poden observar dos paràgrafs diferenciats. El primer copia les dades a la targeta SD i el segon les copia a la pantalla de l'ordinador.

Al principi de tot es pot observar que es crida la funció que s'ha definit anteriorment per escriure l'hora. Per a això, es crida també la funció integrada en la llibreria RTCLib `rtc.now()`, que retorna les diferents variables que s'utilitzen posteriorment per a copiar el temps. Posteriorment, es copien les dades adquirides pel sensor en el següent ordre:

- 1) Concentració PM1.0 [$\mu\text{g}/\text{m}^3$] (Standard)
- 2) Concentració PM2.5 [$\mu\text{g}/\text{m}^3$] (Standard)
- 3) Concentració PM10 [$\mu\text{g}/\text{m}^3$] (Standard)
- 4) Concentració PM1.0 [$\mu\text{g}/\text{m}^3$] (Environmental)
- 5) Concentració PM2.5 [$\mu\text{g}/\text{m}^3$] (Environmental)
- 6) Concentració PM10 [$\mu\text{g}/\text{m}^3$] (Environmental)
- 7) N° partícules mesurades amb diàmetre $>0,3\mu\text{m}$
- 8) N° partícules mesurades amb diàmetre $>0,5\mu\text{m}$
- 9) N° partícules mesurades amb diàmetre $>1,0\mu\text{m}$
- 10) N° partícules mesurades amb diàmetre $>2,5\mu\text{m}$
- 11) N° partícules mesurades amb diàmetre $>5,0\mu\text{m}$
- 12) N° partícules mesurades amb diàmetre $>10\mu\text{m}$

Les tres primeres mesures corresponen als valors de concentració de material particulat que utilitza la fàbrica que crea aquests sensors per a calibrar-los. En un principi doncs, aquests valors no es necessitaran per a res, però ja que el sensor els proporciona i la memòria no és un problema (tal i com es veurà en una de les proves realitzades), s'ha decidit deixar-les.

Les tres següents mesures corresponen als valors de concentració de material particulat que sí són els valors que s'utilitzaran per a mesurar el medi ambient. Aquests tres valors per tant, són els més importants del projecte i els que en un principi es buscava adquirir com a objectiu. A part, el sensor és capaç de donar els valors de PM per a mesures més grans de $2,5\mu\text{m}$ i $1,0\mu\text{m}$, a part de les PM10, paràmetre més utilitzat.

Finalment, el sensor també retorna el comptatge de les partícules que ha realitzat per a fer posteriorment el càlcul de les PM. Tot i que no són valors que normalment s'utilitzin per comparar diferents qualitats de l'aire degut a que aquestes depenen del cabal de l'aire espirat i mesurat, són paràmetres interessants de guardar. I com s'ha esmentat abans, la memòria no és un problema i per tant s'ha optat per incloure les mesures i guardar-les.

6.5. Loop

```
174 // Definició del programa que s'executarà repetidament
175 void loop() {
176   pmsSerial.write(despertar, sizeof(despertar));
177   Serial.println(F("Esperar 30s després de despertar"));
178   delay(30000);
179
180
181   CHECK = false;
182   while (CHECK == false) {
183     arxiu = SD.open("data.txt", FILE_WRITE);
184     if (arxiu) {
185       //ObtenirDataHora(rtc.now());
186       ObtenirDades();
187       arxiu.close();
188     } else {
189       Serial.println(F("Error en obrir l'arxiu"));
190     }
191     delay(100);
192   }
193
194
195   pmsSerial.write(dormir, sizeof(dormir));
196   Serial.println(F("Esperar 30s dormint"));
197   delay(30000);
198 }
```

Il·lustració 21. Codi: Loop del programa

El *loop* és el cos principal del nostre programa. És el fragment de codi que es repetirà de forma continuada i on regularem el funcionament del programa i cridarem les altres funcions.

Primer de tot, s'obliga al motor del sensor a arrencar i començar a fer mesures mitjançant la injecció del conjunt de bytes guardat a la variable “despertar”. Apareixerà un missatge a la pantalla de l'ordinador indicant que el programa esperarà 30 segons per a que les mesures que realitza el sensor arribin a un estat estacionari. El valor de 30 segons és el recomanat pel fabricant. Aquesta espera es realitza mitjançant la comenda `delay()`.

Posteriorment, utilitzarem la variable “CHECK” per a comprovar quan el sensor ha adquirit les dades i s’han guardat a la variable de tipus “struct” mencionada anteriorment i es poden escriure a la targeta SD. Mentre aquesta condició no es compleixi, el codi continuarà intentant adquirir les dades, fins que la comprovació de la comunicació “checksum” sigui correcta.

Seguidament s’obre un arxiu anomenat data.txt en forma d’escriptura i se l’assigna a la variable local “arxiu”. Quan aquest estigui obert i llest, es cridarà la funció d’escriptura de les dades adquirides i es tancarà l’arxiu. En cas que no es pugui obrir l’arxiu, saltarà un error esmentant que hi ha hagut un problema.

Finalment, s’injectarà una altra sèrie de bytes que posaran el sensor en estat inactiu, parant el motor, mitjançant la variable “dormir”. Per últim, el programa esperarà el temps que se li indiqui en l’últim delay(), abans de tornar-se a encendre i prendre una altra mesura. En el codi que hi ha en la imatge superior, el sensor espera 30 segons.

6.6. Funció de comunicació amb el sensor

```
201 boolean readPMSdata(Stream *s) {
202     if (! s->available()) {
203         return false;
204     }
205
206     // Lectura d'un byte cada cop fins que trobem el byte especial d'inici '0x42'
207     if (s->peek() != 0x42) {
208         s->read();
209         return false;
210     }
211
212     // LLegim els 32 bytes següents
213     if (s->available() < 32) {
214         return false;
215     }
216
217     uint8_t buffer[32];
218     uint16_t sum = 0;
219     s->readBytes(buffer, 32);
220
221     // Preparem el 'checksum' de comprovació
222     for (uint8_t i=0; i<30; i++) {
223         sum += buffer[i];
224     }
225
226     uint16_t buffer_u16[15];
227     for (uint8_t i=0; i<15; i++) {
228         buffer_u16[i] = buffer[2 + i*2 + 1];
229         buffer_u16[i] += (buffer[2 + i*2] << 8);
230     }
```

Il·lustració 22. Codi: Funció comunicació amb el sensor (1)

```

231
232 // Passem totes les dades obtingudes a la variable 'struct'
233 memcpy((void *)&data, (void *)buffer_ul6, 30);
234
235 if (sum != data.checksum) {
236     Serial.println("Checksum failure");
237     return false;
238 }
239
240 CHECK = true;
241 return true;
242 }

```

Il·lustració 23. Codi: Funció comunicació amb el sensor (2)

Finalment, s'ha implementat una funció que llegeix la sèrie de bytes que envia el sensor en format hexadecimal, fa la transformació a nombres decimals i assigna cada una d'aquestes dades a la seva variable corresponent. El codi fa ús de la classe stream que ve implementat en el programa de l'Arduino per tal de gestionar aquesta sèrie de dades. Així doncs, i ordenant per paràgrafs:

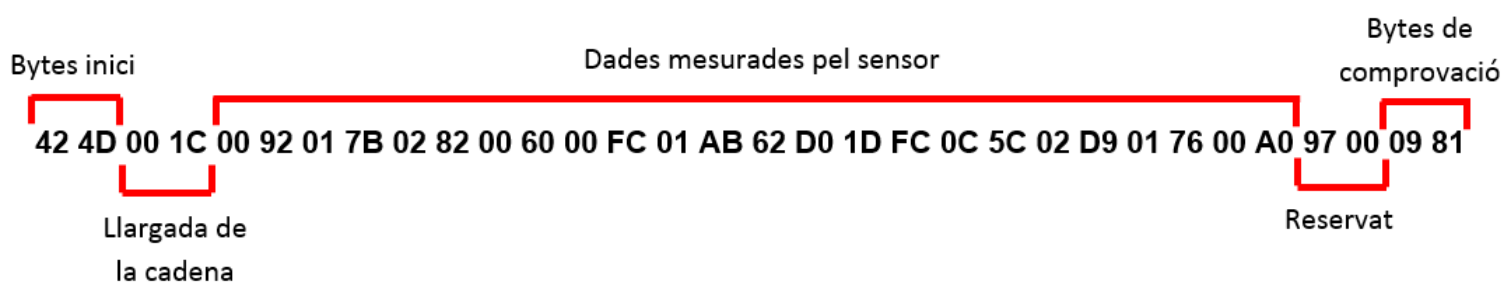
- 1) Línies 202 a 203: Primerament el programa comprova que el canal està obert i que el sensor està transmetent dades de forma continuada. En cas que la condició es compleixi (i per tant la comunicació no estigui disponible), s'atura la funció retornant el valor fals, i la variable CHECK queda com a falsa. Per tant, el fragment de codi dins de Loop on es troba la crida de la funció, es repetiria fins a que aquesta comunicació fos satisfactòria.
- 2) Línies 207 a 210: Posteriorment, el codi llegeix el conjunt de bytes que arriben a través de la comunicació. Busca el byte característic que indica que, a partir d'aquell byte, els següents que venen corresponen a les dades que acaba de mesurar. Aquest és el byte 0x42 (extret del manual del fabricant). En cas que el byte que està llegint no sigui aquest, continua examinant la filera fins que el troba.
- 3) Línies 212 a 219: Un cop trobat el byte característic, el codi llegeix els següents 32 bytes corresponents a les dades mesurades (extret del manual del fabricant). S'emmagatzema aquesta informació en un buffer de 32 bytes i es crea la variable local sum que s'utilitzarà posteriorment per a la comprovació del checksum.
- 4) Línies 222 a 224: En aquestes línies es realitza el càlcul del checksum de la sèrie de dades que s'han transmès, de la forma que especifica el manual del fabricant (la suma de tots els bytes excepte els de verificació). Es guarda aquest valor que s'ha calculat en la variable sum, per la

posterior comprovació amb el byte de comprovació que incorpora la sèrie de dades.

- 5) Línies 226 a 230: Aquí es transforma el conjunt de bytes que estan en hexadecimal a decimal. S'ha de tenir present, tal i com informa el manual del fabricant, que cada una de les dades ve en una parella de bytes (byte low i byte high). Per tant el programa agafa aquesta parella de bytes, obviat els dos bytes d'inici de la cadena, els transforma a decimal i els guarda en un altre buffer, de llargada 15: el primer valor referent a la llargada de la cadena, els 12 següents valors les dades mesurades, el penúltim valor és un espai que es reserva l'empresa per a possibles ampliacions i l'últim valor és el nombre de comprovació.
- 6) Línia 333: Es copien les dades que s'han guardat en el buffer de 15 a cada una de les variables corresponents dins de la variable struct data.
- 7) Línies 235 a 238: es comprova que el byte de comprovació que s'ha enviat i el checksum que s'ha calculat són idèntics. Això confirma que la comunicació de les dades s'ha realitzat correctament i no s'ha perdut cap pel camí.
- 8) Línies 240 a 242: Final de la funció. La variable CHECK passa a valdre verdader i per tant el Loop del programa pot continuar.

6.6.1. Exemple de lectura

Amb l'objectiu d'un major enteniment del que està succeint, es mostra un exemple de la cadena de bytes que llegiria el programa:



Il·lustració 24. Exemple de cadena de bytes enviada pel sensor

Així doncs, la traducció a decimal que realitza el programa, considerant que obvia els dos primers bytes d'inici és:

Nom	Mesura HEX	Mesura DEC
Llargada cadena	001C	28
PM1.0 STD	0092	146
PM2.5 STD	017B	379
PM10 STD	0282	642
PM1.0 ATM	0060	96
PM2.5 ATM	00FC	252
PM10 ATM	01AB	427
Nº Partícules Ø<0,3 µm	62D0	25296
Nº Partícules Ø<0,5 µm	1DFC	7676
Nº Partícules Ø<1,0 µm	0C5C	3164
Nº Partícules Ø<2,5 µm	02D9	729
Nº Partícules Ø<5,0 µm	0176	374
Nº Partícules Ø<10 µm	00A0	160
Reservat	9700	38656
Nº Comprovació	0981	2433

Taula 11. Taula exemple de traducció del missatge rebut pel sensor PMS7003

7. Muntatge

El muntatge del conjunt no ha resultat ser complicat, gràcies a la modularitat dels diferents components que s'utilitzen per projecte que incorporen Arduino. Això implica una reducció dels errors a l'hora d'encarregar els components i la compatibilitat entre aquests. Tot i això, s'ha tingut alguns inconvenients en el sensor PMS7003 que ha obligat adquirir un adaptador adient. També s'ha tingut alguns problemes amb l'alineació dels pins entre components, degut a que alguns han arribat una mica torçats.

7.1. Muntatge pas a pas

Per tal d'exposar el procés de muntatge entre components, es fa un breu llistat de passos a seguir:

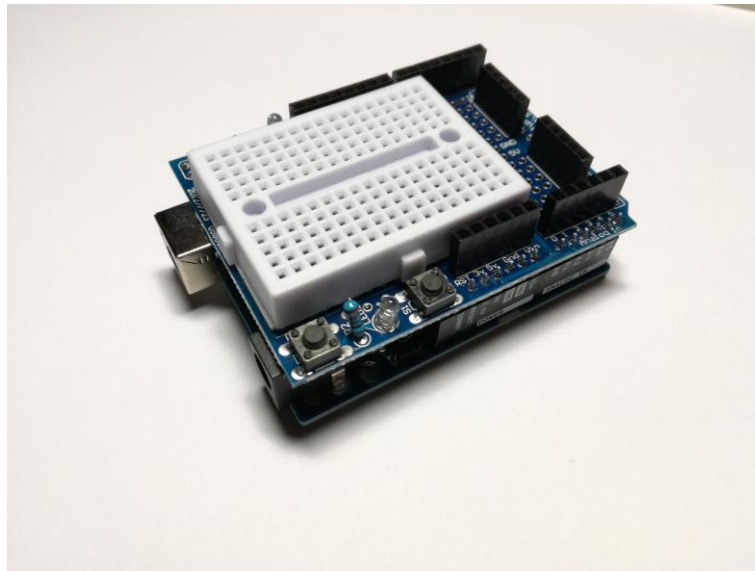
- 1) En primer lloc s'agafa el sensor PMS7003, juntament amb el connector que incorpora i el seu cable. La unió del cable amb el connector s'ha de realitzar amb molta cura, degut a que els cables són molt fins i es podrien arribar a trencar. El final del cable s'uneix a l'adaptador que s'ha adquirit per a facilitar la connexió amb l'Arduino, passant als 2,54mm de secció.



Il·lustració 25. Sensor PMS7003 amb el seu cable

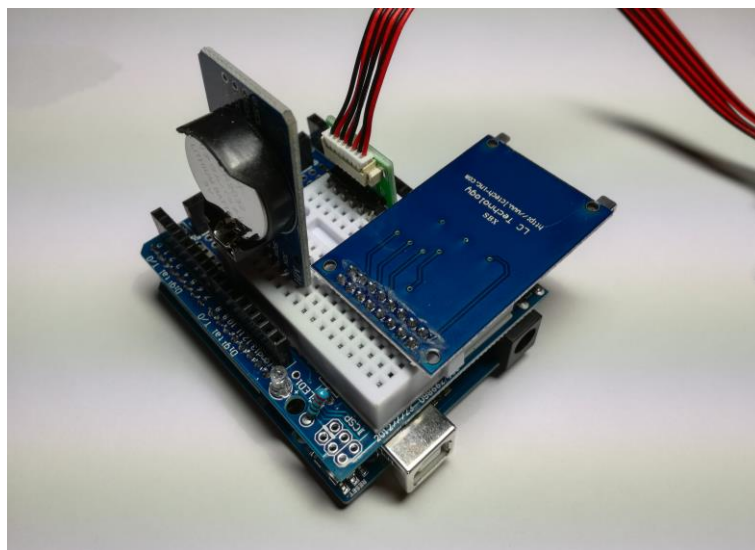
- 2) Posteriorment, es col·loca la ProtoBoard Shield damunt de l'Arduino, fent coincidir tots els pins i vigilant que aquests entrin completament rectes. S'han tingut alguns problemes amb l'alineació dels pins degut a que alguns d'aquests han arribat una mica torçats pel venedor. Les dues

peces han de quedar ben fixades i sense moure's, donant rigidesa al conjunt.



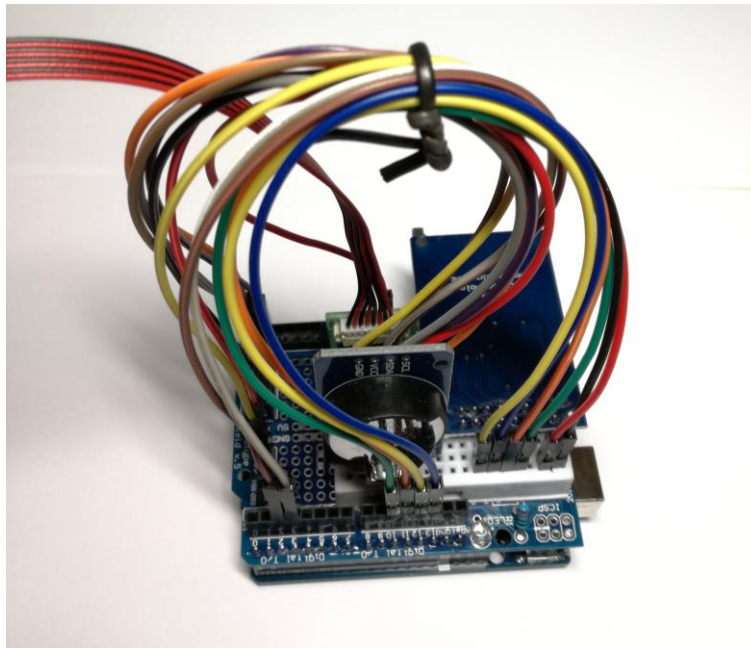
Il·lustració 26. Arduino UNO amb la Protoboard Shield muntada

- 3) S'agafen els diferents components que s'han de connectar (Mòdul SD, Rellotge RTC i Adaptador del sensor) i es col·loquen a la Protoboard Shield. S'ha de deixar la distància suficient entre components com per poder realitzar les connexions correctament, i a la vegada deixin lliure els pins d'entrada i sortida de l'Arduino. Això és especialment important per al Mòdul SD, que és força més gran i es col·loca de forma horitzontal.



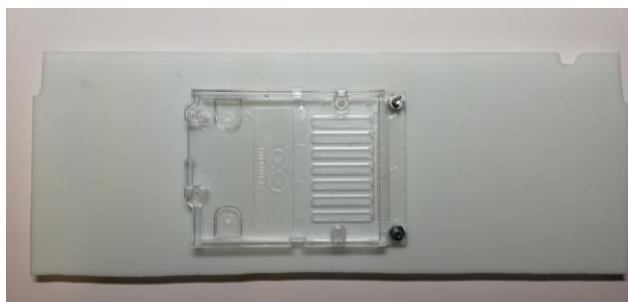
Il·lustració 27. Conjunt de mòduls muntats a l'Arduino

- 4) Mitjançant els cables Jumper, es realitzen les diferents connexions entre els components, d'acord amb l'esquema de connexions que es proporciona més endavant, en el següent apartat. S'agafen els diferents cables amb un brida o similar per a que quedin més ordenats. Ja tenim llest el cor del projecte.



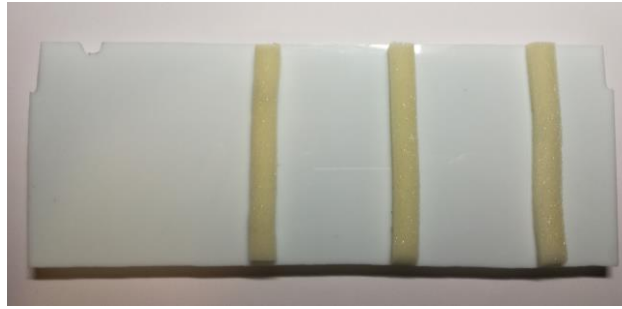
Il·lustració 28. Cablejat entre els diferents mòduls

- 5) Per a la construcció de la caixa, s'ha adquirit una caixa estanca, una platina d'alumini, una altra platina en forma de L d'alumini, un perfil de plàstic en forma de L i una planxa de vidre plàstic llis. També s'ha adquirit dos braços de PVC per tal d'aïllar els forats que es faran posteriorment de la pluja.
- Primer es tallarà un rectangle de longitud igual a la d'un costat de la caixa i amplada igual a l'alçada de la caixa. Aquesta placa anirà ubicada de forma vertical, si tenim la superfície gran de la caixa tocant al terra.
 - Es realitzaran dos forats per a poder collar l'estructura de l'Arduino juntament amb la resta de components (excepte el sensor, que va a part). S'uneix planxa i estructura amb dos cargols i dues femelles.



Il·lustració 29. Placa de plàstic de forma rectangular amb la placa de l'Arduino collada amb cargols

- c. S'afegeixen tires de protecció coincidint amb els caps dels cargols per a no ratllar la bateria.



Il·lustració 30. Tires de protecció al darrera de la placa

- d. Del perfil de plàstic en forma de L, es tallen dos trossos tant llargs com l'alçada de la caixa menys la de la tapa. S'ubica la planxa que acabem de fer de forma vertical i s'enganxa els dos perfils que acabem de tallar a les parets laterals, de forma que facin de guia i de topall per a que el conjunt no caigui. També es deixarà espai per a col·locar la bateria.



Il·lustració 31. Perfil en forma de L que actua com a topall, enganxat a la caixa

- e. Finalment, realitzarem un forat en un dels extrems d'aquesta planxa per a que pugui passar el cable que connecta la bateria amb l'Arduino.



Il·lustració 32. Placa de plàstic juntament amb l'Arduino, col·locat dins la caixa

- f. Es continuarà realitzant dos forats en un dels costats de la caixa, convenientment grans per a què no interfereixin en el flux d'aire absorbit pel sensor. S'ha de vigilar de no ubicar-los massa a baix ni massa a dalt, per a que no hi hagi interferències.



Il·lustració 33. Forats realitzats a la caixa, on es pot observar el sensor a l'altra banda, i els braços de PVC que s'instal·laran posteriorment

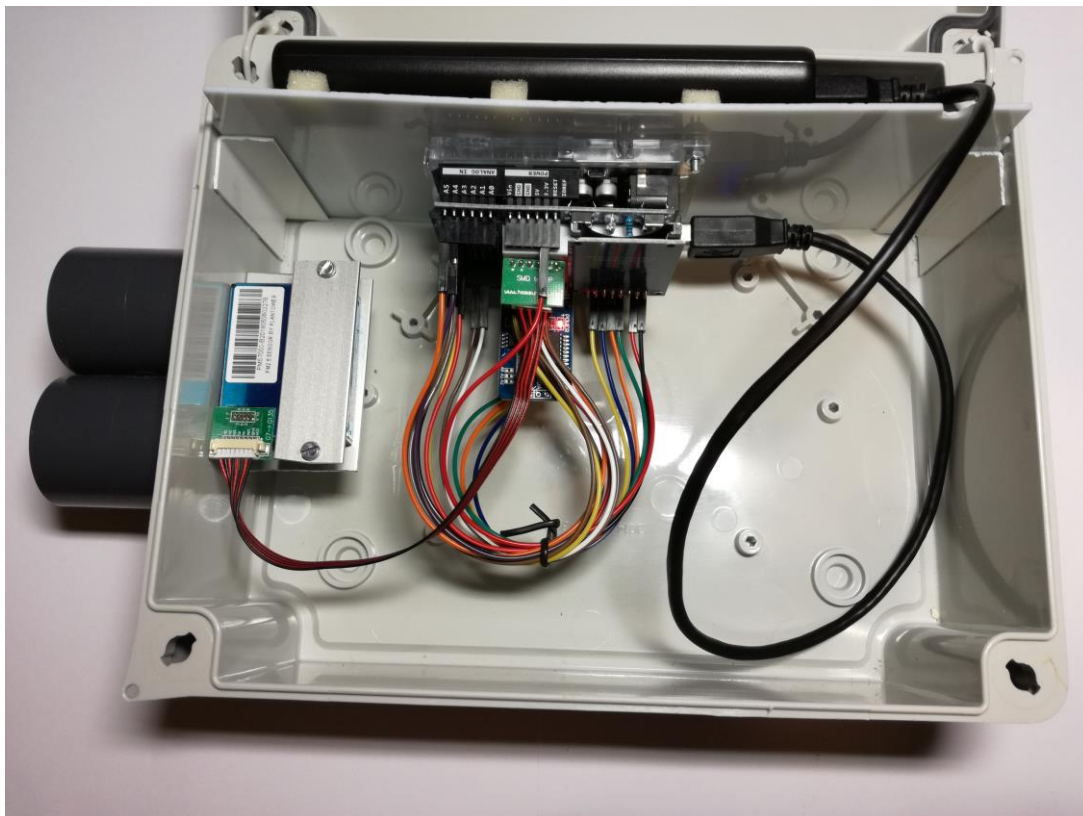
- g. Posteriorment, es tallarà les dues platines d'alumini (una plana i l'altra en forma de L) de forma que sobresurtin una mica del sensor, ubicades de forma longitudinal amb aquest. Es marcarà la ubicació on aniran els cargols i es foradarà. L'objectiu és el de fixar el sensor fent pinça amb les dues platines.
- h. Col·locant el conjunt de platines que acabem de muntar de forma que els forats del sensor quedin alineats amb els de la caixa, marquem per fer dos forats més (prop dels dos més grans que hem

fet posteriorment), per tal de fixar la platina en forma de L a la caixa. Es fan els forats i es colla el conjunt amb dos cargols.



Il·lustració 34. Instal·lació i fixació del sensor a la caixa

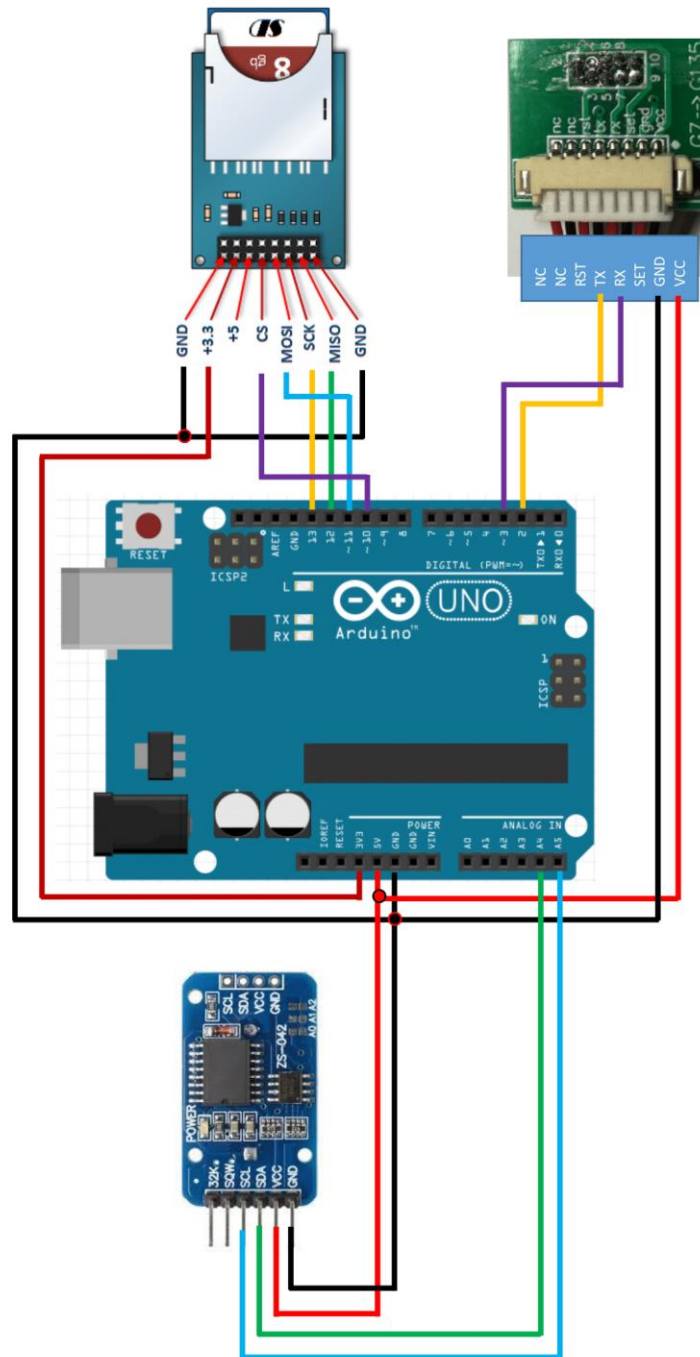
- i. Finalment, s'enganxen els dos braços de PVC per l'exterior, cada un d'ells abastant un dels forats grans i un dels cargols.
- 6) Per últim col·loquem la planxa amb l'Arduino, la bateria i el cable que els uneix dins la caixa.



Il·lustració 35. Construcció de la caixa i instal·lació de l'Arduino completada

7.2. Esquema de connexions

L'esquema de connexions dels diferents mòduls amb els pins de l'Arduino que s'ha utilitzat per aquest projecte són els següents:



Il·lustració 36. Esquema de connexions entre l'Arduino i els diferents mòduls

8. Proves

Durant el projecte s'han realitzat diferents proves per tal de poder calibrar el consum, calcular la duració de funcionament amb una capacitat de bateria determinat, comprovar la fiabilitat del sensor comparant les mesures preses, i altres proves per a comprovar que el funcionament del programa fos el correcte. Les proves realitzades es llisten aquí.

8.1. Capacitat de la targeta SD

Una de les incògnites a l'hora de plantejar el projecte és quina capacitat hauria de tenir la targeta SD per tal de poder emmagatzemar totes les dades que s'anessin adquirint sense haver d'anar a bolcar-les massa sovint.

Per a trobar aquest número, es va realitzar una prova on el sensor prenia dades i les emmagatzemava a cada segon. Aquesta és la freqüència mínima, aproximadament, a la que el sensor pot agafar dades. Per tant, s'ha realitzat la prova sota les condicions més crítiques. S'ha dut a terme la prova durant 24h, utilitzant una targeta de 16GB, a priori suficient per a emmagatzemar totes les dades.

Un cop realitzada la prova, s'ha observat la capacitat ocupada del document de text que emmagatzema les dades. El valor d'aquest ha estat de 4,68MB, havent pres un total de pràcticament 100.000 dades. Això implica que, mantenint la targeta utilitzada, es podria haver pres dades durant 3.500 dies, equivalent a 9 anys i mig.

Resulta evident doncs que la capacitat d'emmagatzemament de dades no és un problema per al projecte. Així doncs, una targeta d'1GB es considera suficient per al projecte, permetent una mica més de 6 mesos de presa de dades contínua.

8.2. Capacitat de la bateria

L'altre de les incògnites importants que s'havia de dimensionar és la capacitat de la bateria. Per tal de poder fer el correcte dimensionament, era necessari mesurar el consum de tot el conjunt durant la presa de dades i durant el mode inactiu.

Per a aconseguir això, s'ha adaptat un cable USB. S'ha tallat l'aïllament d'aquest, deixant al descobert els dos cables de contínua. S'ha tallat el de massa i s'ha intercalat entre ells un multímetre, per tal de mesurar el corrent consumit durant

cada una de les fases. S'ha comprovat que mentre l'Arduino executa el codi i el sensor pren mesures de forma continuada, el consum del conjunt és d'aproximadament 150mAh. Quan el sensor deixa de prendre mesures i para el ventilador petit que incorpora, aquest consum disminueix a 70mAh, provinents de l'Arduino. Per tant, el consum mig dependrà de quanta estona està el sensor en estat inactiu (entre mesures). Suposant 30 segons d'escalfament, prendre la mesura i 270 segons amb el sensor parat (temps de cicle de 5 minuts), això implicaria un consum d'uns 78mAh, un valor prou raonable. Agafant aquests valors de referència, una bateria de 10.000mAh duraria aproximadament 5 dies i una de 25.000mAh, 12 dies. Aquests són valors de referència i depenen en gran mesura de la qualitat de la bateria, de la capacitat eficaç d'aquesta, de la temperatura ambient i del consum real del conjunt Arduino-Sensor.

Per al tipus de bateria a utilitzar, s'han tingut en compte diferents possibilitats: bateries de plom, piles de 9V, bateries per a carregar mòbils o Power Banks. La limitació de voltatge màxim que admet Arduino (per sobre de 12V, tot i que no és gens recomanable passar d'aquest valor en estat estacionari) ha limitat en gran mesura l'elecció del tipus de bateria. El pes també ha estat un factor decisiu, degut a que la caixa ha de ser fàcilment transportable, així com la càrrega de les bateries. Per tant, finalment s'ha elegit utilitzar una bateria d'ió liti tipus Power Bank, ja que subministren 5V molt estables, pesen poc i actualment existeixen bateries amb una capacitat de càrrega prou decent.

8.3. Comprovació que els braços de PVC incorporats no modifiquen les mesures

Per tal de comprovar que la incorporació dels braços de PVC a la caixa no interfereixen en la qualitat de les mesures, s'ha dut a terme una prova amb una rèplica del projecte.

S'han disposat les dues caixes en el mateix lloc i s'han activat les mesures en el mateix moment. El que distingeix una caixa de l'altra, és que en la primera el sensor està a dins i la caixa tancada, tal i com està previst

l'ús del sensor; i en la segona el sensor penja per fora de la caixa, estant aquesta oberta. Es deixaran ambdues prenent dades durant un dia sencer.



Il·lustració 37. Dues caixes emprades per la prova dels braços: la de l'esquerra amb el sensor per dins i la de la dreta amb el sensor per fora

S'ha de tenir en compte que les condicions de mesura no són les ideal, ja que el flux d'aire que passa per cada un dels sensors a cada moment no és el mateix, així com la composició exacta. A més, la variabilitat entre sensors és relativament alta, degut al seu cost tant baix. S'ha demostrat en altres estudis que la variació en la mesura és bastant alta depenent de la temperatura i la humitat de l'aire.

Els resultats de la prova mostren que el desviament de les dades entre un i altre sensor no destaca en gran mesura i es manté dins dels límits normals, sense mostrar puntes de variació molt exagerades. Amb això, es pot afirmar que l'impacte de l'ús dels braçals de PVC per a protegir de la pluja no té una implicació més gran en la mesura que la d'altres factors esmentats en el paràgraf anterior.

Es poden trobar els gràfics de la comparació de dades entre els dos sensors a l'Annex 2.

8.4. Comparació de mesura amb una estació de control de qualitat de l'aire de la Generalitat de Catalunya

Com s'ha esmentat anteriorment en la memòria, la Generalitat de Catalunya va impulsar la construcció i implementació d'una xarxa de control de la qualitat de l'aire anomenada Xarxa de Vigilància i Previsió de la Contaminació Atmosfèrica (XVPCA), amb l'objectiu de mantenir un seguiment dels contaminants. Aquestes estacions recullen les dades mesurades en un interval d'una hora. Posteriorment es guarden en una base de dades accessible per qualsevol persona per a poder-les consultar.

Així doncs, aprofitant aquesta oportunitat, s'ha portat el sensor construït al costat d'una d'aquestes estacions que mesuren també les PM10 de l'aire, i s'ha deixat realitzant mesures per comparar-les posteriorment. L'objectiu d'aquesta prova és el de comprovar que els sensors de baix cost poden donar una mesura prou fiable com per tenir-los en compte i que, a grans trets, es poden fer servir per mantenir un seguiment prou segur de la contaminació de la zona.

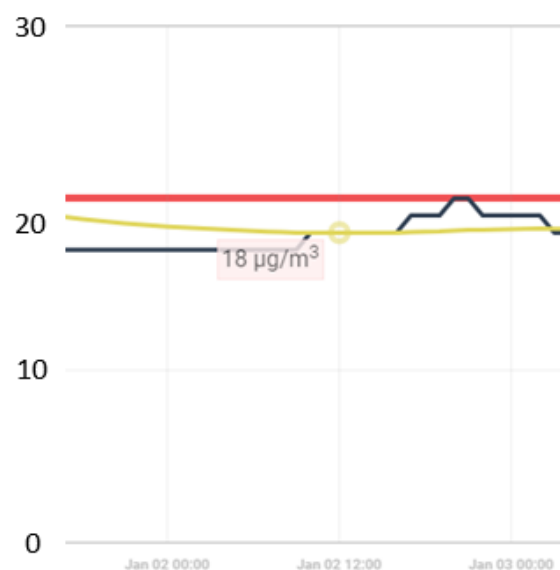
L'estació escollida es troba ubicada a Barcelona, a la Plaça de Gal·la Placídia, al barri de Gràcia. Les dades d'aquesta són:

ESTACIÓ DE MESURA	
Nom estació	Barcelona (Gracia – Sant Gervasi)
Codi EOI	8019044
Codi província	8
Codi municipi	19
Codi estació	44
Municipi	Barcelona
Latitud	41,398724
Longitud	2,15339888
Altitud	57

Taula 12. Informació de l'estació amb la que es comparen les dades adquirides

Les dades les ha adquirit el professor i director d'aquest treball, el senyor Jesús Andrés Álvarez Florez fent ús del sensor construït, degut a l'accessibilitat de l'estació. Donat que les mesures que realitza l'estació són en intervals d'una hora, i que l'objectiu últim d'aquest treball no és el d'analitzar el grau de fiabilitat de les mesures, només s'han pres mesures al voltant d'una d'aquestes hores.

El dia i hora són el 2 de gener de 2020, a les 12h del migdia. El valor de PM10 mesurat va ser de 18 $\mu\text{g}/\text{m}^3$.



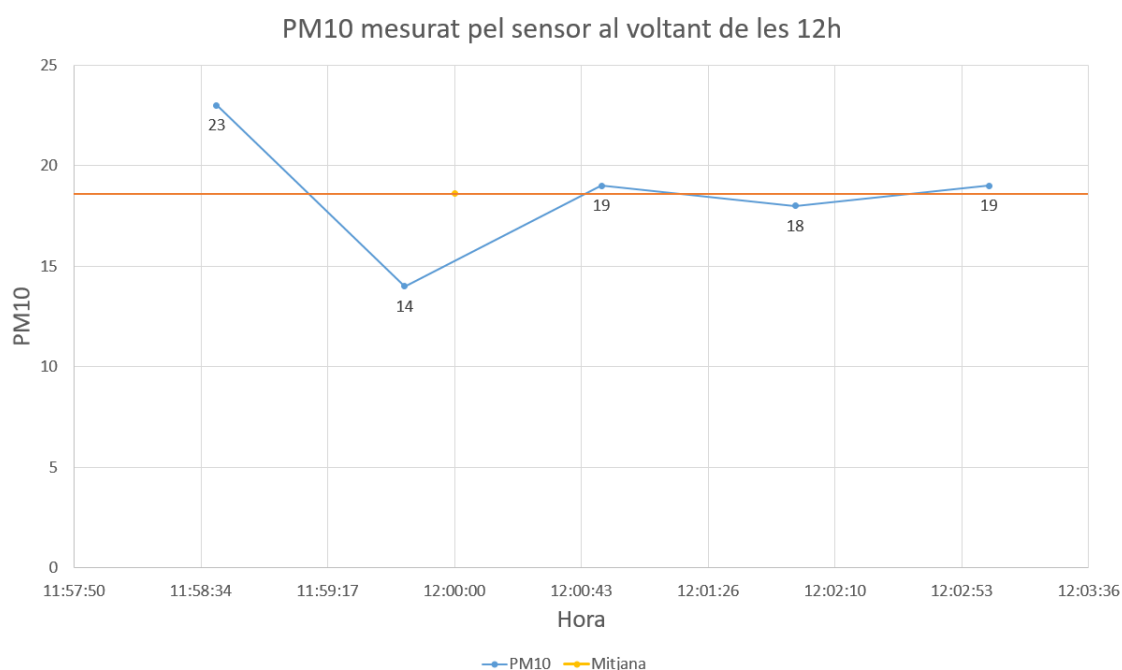
Il·lustració 38. Gràfica de dades obtingudes per l'estació de mesura

Aquesta informació ha estat extreta directament de la pàgina web disposada per a la visualització de les dades que es recullen, a www.contaminacio.cat. La línia negra indica el valor horari mesurat, la groga és la mitjana i la vermella el valor límit de la mitjana anual establert per la legislació europea ($40.0 \mu\text{g}/\text{m}^3$) i de l'Organització Mundial de la Salut ($20.0 \mu\text{g}/\text{m}^3$).

Els valors que s'han recollit utilitzant el sensor construït al voltant d'aquesta hora són:

Dia	Hora	PM1.0	PM2.5	PM10	D0.3	D0.5	D1.0	D2.5	D5	D10
02/01/2020	11:58:39	14	21	23	2145	606	128	8	3	1
02/01/2020	11:59:43	11	14	14	1626	453	78	0	0	0
02/01/2020	12:00:50	13	18	19	1914	546	116	3	3	1
02/01/2020	12:01:56	12	14	18	1785	498	70	5	5	1
02/01/2020	12:03:02	11	16	19	1803	505	78	7	4	2

Taula 13. Dades obtingudes pel sensor construït al voltant de les 12h



Il·lustració 39. Gràfica de dades obtingudes pel sensor construït al voltant de les 12h

Es pot observar que les dades adquirides són molt properes al valor de referència de l'estació. Puntualment es poden produir pics i valls degut a les diferents variables que intervenen, com poden ser el vent i la no-homogeneïtat de la distribució de la contaminació. S'observa, però, que la mitjana entorn d'aquest punt, de $18,6 \mu\text{g}/\text{m}^3$, és prou propera a la de l'estació.

Conclusions

El projecte s'iniciava amb l'objectiu de construir, amb el mínim cost possible, un dispositiu de mesura del material suspès a l'atmosfera amb l'ànim de controlar un dels paràmetres de la qualitat de l'aire: les PM10. Aquest és la resposta a l'auge dels últims anys per la preocupació sobre el medi ambient i l'increment de la contaminació a les ciutats, propiciant que persones no professionals, dissenyin i construeixin el seu propi sistema de control de l'aire.

Optant per un model basat en el popular controlador Arduino, s'ha aconseguit la fita de mantenir el cost del projecte sota uns valors controlats i assequible per a que el ciutadà mig pogués construir-lo. L'elecció d'aquest controlador també obre les portes a futures modificacions i ampliacions, degut a la modularitat del sistema i la quantitat de material disponible, en forma de mòduls. La construcció de la caixa s'ha dissenyat per a deixar espai a aquestes.

S'ha aconseguit escriure un programa que controli els diferents mòduls, adquirint les dades que envia el sensor de PM10 i emmagatzemar-les en una targeta SD, per a una posterior transferència a un ordinador per poder-les analitzar. També s'ha aconseguit que el sensor quedi en un estat inactiu mentre no mesura, parant el motor que incorpora i estalviant així energia de la bateria, element crític degut a l'alt consum inherent als controladors Arduino.

Així doncs, la construcció i programació del conjunt ha estat satisfactòria. Tot i això, és possible que es pugui millorar el codi programat, fent-lo més eficient i robust a alteracions externes. No s'ha pogut comprovar si existeix o no variació de la mesura al llarg del temps, degut a la contaminació acumulada en els conductes d'aspiració de l'aire del sensor, així com altres factors de manteniment del conjunt. Com a tal, s'hauria de realitzar un estudi posterior per tal de garantir i proposar el temps útil del sensor.

Bibliografia

- Arduino. (s.f.). *Arduino UNO Rev3*. Obtenido de <https://store.arduino.cc/arduino-uno-rev3>
- Castell, N., Dauge, F., Schneider, P., Vogt, M., Lerner, U., Fishbain, B., . . . Bartonova, A. (2017). Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates? *Environment International* 99, 293-302.
- Departament de Territori i Sostenibilitat. (s.f.). *Avaluació de la qualitat de l'aire*. Obtenido de http://mediambient.gencat.cat/ca/05_ambits_dactuacio/atmosfera/qualitat_de_laيرة/avaluacio/
- Departament de Territori i Sostenibilitat. (s.f.). *Índex català de la qualitat de l'aire (ICQA)*. Obtenido de http://mediambient.gencat.cat/ca/05_ambits_dactuacio/atmosfera/qualitat_de_laيرة/avaluacio/icqa/
- Departament de Territori i Sostenibilitat. (s.f.). *Normativa catalana qualitat de l'aire*. Obtenido de http://mediambient.gencat.cat/es/05_ambits_dactuacio/atmosfera/qualitat_de_laيرة/valors_legislats_de_la_qualitat_de_laيرة/
- Generalitat de Catalunya. (s.f.). *Xarxa de vigilància i Previsió de la Contaminació Atmosfèrica a Catalunya*. Obtenido de http://www.gencat.cat/mediamb/qaire/web/mapa_nou_2005.htm
- Ministerio para la Transición Ecológica. (s.f.). *Normativa española calidad del aire*. Obtenido de <https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/calidad-del-aire/normativa/normativa-espanola.aspx>
- Ministerio para la Transición Ecológica. (s.f.). *Normativa europea calidad del aire*. Obtenido de <https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/calidad-del-aire/normativa/normativa-europea.aspx>
- Portal Jurídic de Catalunya. (s.f.). *Llei 22/1983*. Obtenido de https://portaljuridic.gencat.cat/ca/pjur_ocults/pjur_resultats_fitxa/?action=fitxa&documentId=15802
- Vazquez, D. G. (s.f.). *El País*. Obtenido de https://elpais.com/ccaa/2014/01/25/catalunya/1390674525_016528.html
- Wikipedia. (s.f.). *Benceno*. Obtenido de <https://es.wikipedia.org/wiki/Benceno>
- Wikipedia. (s.f.). *Dióxido de azufre*. Obtenido de https://es.wikipedia.org/wiki/Di%C3%B3xido_de_azufre
- Wikipedia. (s.f.). *Óxidos de nitrógeno*. Obtenido de https://es.wikipedia.org/wiki/%C3%93xidos_de_nitr%C3%B3geno
- Wikipedia. (s.f.). *Oxocarbono*. Obtenido de <https://es.wikipedia.org/wiki/Oxocarbono>
- Wikipedia. (s.f.). *Ozono*. Obtenido de <https://es.wikipedia.org/wiki/Ozono>
- Wikipedia. (s.f.). *Partículas en suspensión*. Obtenido de https://es.wikipedia.org/wiki/Part%C3%ADculas_en_suspensi%C3%B3n